

**IMPLEMENTASI TEKNIK ENKODING DIGITAL PEMBACAAN
SENSOR ULTRASONIK UNTUK MEMETAKAN KEPUTUSAN
AKSI ROBOT QUADRUPEL**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

Disusun oleh:
Oggy Setiawan
NIM: 145150300111072



**PROGRAM STUDI TEKNIK KOMPUTER
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

IMPLEMENTASI TEKNIK ENKODING DIGITAL PEMBACAAN SENSOR ULTRASONIK
UNTUK MEMETAKAN KEPUTUSAN AKSI ROBOT QUADRUPEL

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
Memperoleh gelar Sarjana Teknik

Disusun Oleh:
Oggy Setiawan
NIM : 145150300111072

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Juli 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Dahnial Syauqy, S.T., M.T., M.Sc.
NIK : 20160788704231002

Wijaya Kurniawan, S.T., M.T.
NIP : 198201252015041002



Mengetahui
Ketua Jurusan Teknik Informatika

Irfan Astoto Kurniawan, S.T., M.T., Ph.D
NIP : 197105182003121001

PERNYATAAN ORISINILITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 10 Juli 2018



Oggy Setiawan

NIM : 145150300111072

KATA PENGANTAR

Segala puji syukur penulis ucapkan kepada Alla SWT karena berkat rahmat dan hidayah-Nya penulis dapat menyelesaikan penelitian dan Laporan skripsi yang berjudul **Implementasi Teknik Enkoding Digital Pembacaan Sensor Ultrasonik Untuk Memetakan Keputusan Aksi Robot Quadruped** untuk memperoleh gelar Sarjana Komputer.

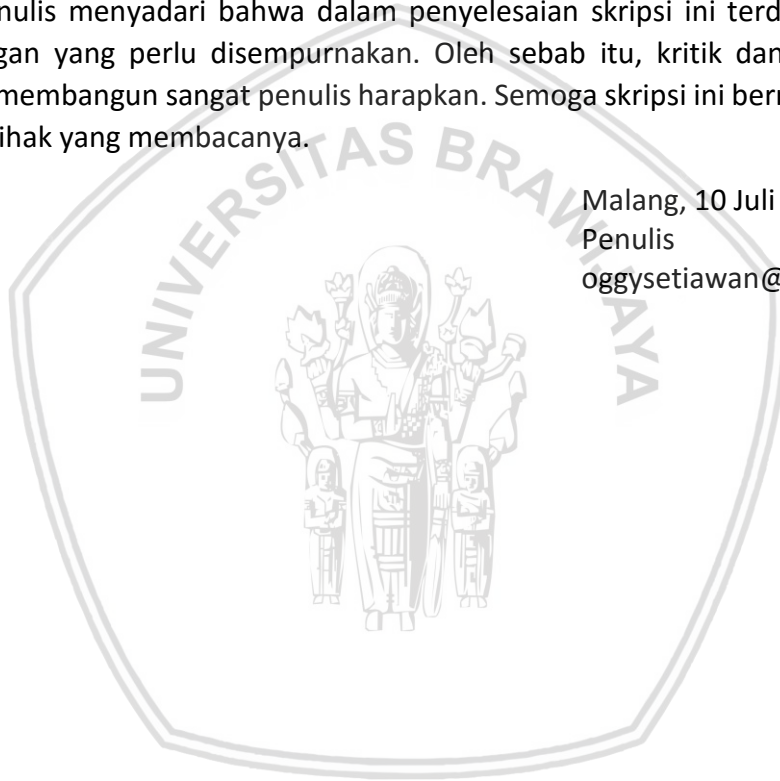
Tidak sedikit hambatan yang penulis hadapi dalam pelaksanaan dan penyusunan laporan skripsi ini. Namun penulis menyadari bahwa dalam penyusunan laporan ini tidak lepas dari bantuan baik motivasi, doa, dan bimbingan dari berbagai pihak, sehingga kendala-kendala yang penulis hadapai dapat teratasi. Penghargaan, rasa hormat dan terimakasih yang sebesar-besarnya penulis sampaikan kepada:

1. Kedua orang tua penulis yaitu Bapak Mokid dan Ibu Pujiwati, serta Gita Fitri selaku saudara kandung penulis yang tiada henti memberikan semangat dan dorongan baik berupa do'a maupun berupa materi selama penulis melakukan penelitian.
2. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Informatika
3. Bapak Sabriansyah Rizqika S.T., M.Eng. selaku ketua Program Studi Teknik Komputer
4. Bapak Dahnial Syauqy S.T., M.T., M.Sc. selaku pembimbing I yang telah memberikan bimbingan dan arahan selama pengerjaan skripsi.
5. Bapak Wijaya Kurniawan, S.T., M.T. selaku pembimbing II yang telah memberikan bimbingan dan arahan selama pengerjaan skripsi.
6. Bapak Ibu Dosen Program Studi Teknik Komputer yang telah memberikan ilmu selama berkuliah.
7. Almarhumah Ibu Mulyati selaku nenek penulis, dan seluruh keluarga penulis yang tiada henti memberikan motivasi dan semangat serta doa sehingga penulis dapat menyelesaikan skripsinya.
8. Teman-Teman D'gengz Ida Y, Intan F, Yongki P, Linda Silvy P, Muzammilatul J, Lita N, Putri A, Syahriel D, Adrian Mitra P, Viramuda Burhan, Ilham N, dan Tezza R yang telah mengisi hari-hari penulis untuk menemani dan menyemangati dalam menyelesaikan penulisan skripsi ini.

9. Teman-teman penulis yang berasal dari Pekanbaru Valencia Ayu, Vinoza Ayu, dan yang tidak bisa di sebutkan satu-persatu yang selalu memotivasi, memberikan doa dan semangat kepada penulis untuk segera menyelesaikan penulisan skripsi ini.
10. Hendriawan, HIMATEKKOM, Djarum Beasiswa Plus, teman-teman Fakultas Ilmu Komputer, dan teman-teman Teknik Komputer 2014 yang selalu memotivasi penulis.
11. Seluruh pihak yang tidak dapat di sebutkan satu persatu yang telah berperan dalam penyelesaian skripsi ini.

Penulis menyadari bahwa dalam penyelesaian skripsi ini terdapat banyak kekurangan yang perlu disempurnakan. Oleh sebab itu, kritik dan saran yang bersifat membangun sangat penulis harapkan. Semoga skripsi ini bermanfaat bagi semua pihak yang membacanya.

Malang, 10 Juli 2018
Penulis
oggysetiawan@gmail.com



ABSTRAK

Perkembangan Teknologi telah menciptakan berbagai kemajuan teknologi, terutama dibidang robotika. Robot Berkaki memiliki beberapa keuntungan yaitu mampu bergerak di daerah yang halus astau kasar, bergelombang dan tidak rata atau datar, memanjat tangga, menghindar dan melangkah di atas halangan. Salah satu masalah dalam merancang kemampuan robot adalah sistem lokalisasi robot yaitu untuk mengetahui posisi pada suatu waktu tertentu. Teknik enkoding digital adalah rangkaian kombinasional yang menghasilkan spesifikasi pada keluarannya seperti biner sebagai respons terhadap satu atau lebih masukan yang aktif yang berguna mengimperesi data dan dapat dibangun dari gerbang logika AND dan OR sederhana. Proses teknik enkoding digital dimulai dengan pembuatan tabel kebenaran, rumus teknik enkoding digital, dan penyederhanaan menggunakan K-MAP. Mengimplementasikan teknik enkoding digital menggunakan 8 buah sensor ultrasonik untuk memetakan keputusan 8 aksi gerak robot quadruped. Pengujian dilakukan terhadap tiga hal yaitu, akurasi sensor ultrasonik HC-SR04, waktu komputasi teknik enkoding digital, dan akurasi hasil klasifikasi teknik enkoding digital. Pengujian tersebut didapatkan akurasi aksi gerak robot dengan mengimplementasikan teknik enkoding digital sebesar 98% dengan rata-rata waktu komputasi sekitar 0,16 detik saat salah satu sensor diberi halangan sejauh 7 cm dan 0,17 detik saat salah satu sensor diberi halangan sejauh 20 cm.

Kata Kunci : Robot, Robot Quadruped, Teknik Enkoding Digital, Sensor Ultrasonik HC-SR04

ABSTRACT

Technology development has created various kinds of technology improvement, especially in robotics. Legged-robots has several advantages; they can move in both smooth and rough areas, wavy, uneven or flat, climb the stairs, avoid and step in the distractions. One of the problem in designing a robot's ability is the robot's localization system to know the position in a certain time. Digital encoding technique is a series of combination that produces specific output such as binary as a response to one or more active input that is used to simplify data and can be built from a simple logic gate AND and OR. The digital encoding process start from the making of truth table, digital encoding technique formula, and simplification by K-MAP. Implementing digital encoding technique using 8 ultrasonic sensors to map 8 motor actions of the quadruped robot. Testing is done in three things; the accuracy of ultrasonic sensor HC-SR04, digital encoding computing time, and the accuracy of classification result in digital encoding technique. From the test we can get robot's motor action accuracy by implementing digital encoding technique as much as 98% with the average computing time is around 0.16 seconds when one of the sensors is given a 7 cm incline and 0,17 seconds when one of the sensors is given a 20 cm incline.

Keywords: Robot, Robot quadruped, digital encoding technique, ultrasonic sensor HC-SR04

DAFTAR PUSTAKA

PENGESAHAN	ii
PERNYATAAN ORISINILITAS.....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR PUSTAKA.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Batasan Masalah	3
1.6 Sistematika Pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Tinjauan Pustaka	4
2.1.1 Desain dan Implementasi Kendali Cerdas untuk Robot <i>Quadpod</i> (Berkaki Empat) Studi Kasus Robot Pemadam Api	4
2.1.1 Algoritma Fuzzy Logic dan <i>Wallfollower</i> pada Sistem Navigasi Robot <i>Hexapod</i> Berbasis Mikrokontroler AVR	4
2.2 Dasar Teori.....	5
2.2.1 Gerbang Logika.....	5
2.2.2 Teknik Enkoding Digital	9
2.2.3 Robot.....	9
2.2.4 Arduino Mega 2560.....	10
2.2.5 Sensor Ultrasonik HC-SR04	11
2.2.6 OpenCM9.04	12
2.2.7 Board Ekspansi OpenCM485	13

2.2.8	Servo Dynamixel AX-12A.....	13
2.2.9	Arduino IDE	14
2.2.10	FreeRTOS	15
BAB 3 METODOLOGI		16
3.1	Alur Metodologi Penelitian	16
3.2	Studi Literatur.....	17
3.3	Analisis Kebutuhan	17
3.4	Pengambilan Data	18
3.5	Perancangan Sistem	18
3.6	Implementasi Sistem.....	18
3.7	Pengujian dan Analisis.....	18
3.8	Kesimpulan	19
BAB 4 REKAYASA KEBUTUHAN.....		20
4.1	Gambaran Umum Sistem	20
4.2	Analisis Kebutuhan Sistem	20
4.2.1	Kebutuhan Fungsional.....	20
4.2.2	Kebutuhan Non Fungsional.....	21
4.3	Batasan Desain Sistem.....	22
BAB 5 PERANCANGAN DAN IMPLEMENTASI		23
5.1	Perancangan Sistem	23
5.1.1	Perancangan Robot Quadruped	23
5.1.2	Perancangan Perangkat Keras	24
5.1.3	Perancangan Perangkat Lunak.....	26
5.2	Implementasi Sistem	32
5.2.1	Implementasi Robot <i>Quadruped</i>	32
5.2.2	Implementasi Perangkat Keras	33
5.2.3	Implementasi Perangkat lunak	34
BAB 6 PENGUJIAN DAN ANALISIS.....		38
6.1	Pengujian Sensor Ultrasonik HC-SR04.....	38
6.1.1	Tujuan Pengujian.....	38
6.1.2	Prosedur Pengujian	38

6.1.3	Hasil dan Analisis Pengujian	39
6.2	Pengujian Waktu Komputasi Teknik Enkoding Digital.....	43
6.2.1	Tujuan Pengujian.....	43
6.2.2	Prosedur Pengujian	43
6.2.3	Hasil dan Annalisis Pengujian.....	44
6.3	Pengujian Akurasi Hasil Klasifikasi Teknik Enkoding Digital	47
6.3.1	Tujuan Pengujian.....	47
6.3.2	Prosedur Pengujian	47
6.3.3	Hasil dan Annalisis Pengujian.....	47
BAB 7	PENUTUP	49
7.1	Kesimpulan	49
7.2	Saran.....	49
DAFTAR	PUSTAKA.....	51
LAMPIRAN A	TABEL KEBENARAN	52



DAFTAR TABEL

Tabel 2.1 Tabel Kebenaran Gerbang <i>AND</i>	5
Tabel 2.2 Tabel Kebenaran Gerbang <i>OR</i>	6
Tabel 2.3 Tabel Kebenaran Gerbang <i>NOT</i>	6
Tabel 2.4 Tabel Kebenaran Gerbang <i>NAND</i>	7
Tabel 2.5 Tabel Kebenaran Gerbang <i>NOR</i>	8
Tabel 2.6 Tabel Kebenaran Gerbang <i>X-OR</i>	8
Tabel 2.7 Tabel Kebenaran Gerbang <i>X-NOR</i>	9
Tabel 2.8 Spesifikasi Arduino Mega 2560	11
Tabel 2.9 Parameter Sensor Ultrasonik HC-SR04	12
Tabel 5. 1 Tabel Pin Skematik	26
Tabel 5. 2 Penamaan Arah Gerak Robot.....	29
Tabel 5. 3 Tabel Input yang memiliki Output A	30
Tabel 5. 4 K-MAP Pergerakan Robot A.....	31
Tabel 5. 5 Inisialisasi Library.....	34
Tabel 5. 6 Kode Program Implementasi Kode Program Pengambilan Data Sensor	35
Tabel 5. 7 Kode Program Inisialisasi Variabel Teknik Enkoding Digital.....	35
Tabel 5. 8 Kode Program Implementasi Teknik Enkoding Digital	36
Tabel 6. 1 Hasil Pengujian Sensor Ultrasonik HC-SR04 1	40
Tabel 6. 2 Hasil Pengujian sensor Ultrasonik HC-SR04 2	40
Tabel 6. 3 Hasil Pengujian Sensor Ultrasonik HC-SR04 3	40
Tabel 6. 4 Hasil Pengujian Sensor Ultrasonik HC-SR04 4	40
Tabel 6. 5 Hasil Pengujian Sensor Ultrasonik HC-SR04 5	41
Tabel 6. 6 Hasil Pengujian Sensor Ultrasonik HC-SR04 6	41
Tabel 6. 7 Hasil Pengujian Sensor Ultrasonik HC-SR04 7	41
Tabel 6. 8 Hasil Pengujian Sensor Ultrasonik HC-SR04 8	42
Tabel 6. 9 Hasil Pengujian Semua Sensor Ultrasonik	42
Tabel 6. 10 Kode Program Menghitung Waktu Komputasi	44
Tabel 6. 11 Hasil Pengujian Waktu Komputasi Dengan Salah satu sensor berjarak 7cm.....	45

Tabel 6. 12 Hasil Pengujian Waktu Komputasi dengan Salah Satu Sensor Berjarak 20cm	45
Tabel 6. 13 Hasil Pengujian Teknik Enkoding Digital.....	47



DAFTAR GAMBAR

Gambar 2.1 Simbol Gerbang <i>AND</i>	5
Gambar 2.2 Simbol Gerbang <i>OR</i>	6
Gambar 2.3 Simbol Gerbang <i>NOT</i>	6
Gambar 2.4 Simbol Gerbang <i>NAND</i>	7
Gambar 2.5 Simbol Gerbang <i>NOR</i>	7
Gambar 2.6 Simbol Gerbang <i>X-OR</i>	8
Gambar 2.7 Simbol Gerbang <i>X-NOR</i>	8
Gambar 2. 8 Contoh Tabel Kebeneran Empat Masukan Dua Keluaran	9
Gambar 2.9 Arduino Mega Pin Mapping	10
Gambar 2.10 HC-SR04	11
Gambar 2.11 <i>Layout</i> dan <i>Pin Mapping</i> OpenCM9.04	12
Gambar 2.12 <i>Layout</i> OpenCM485 dengan OpenCM9.04	13
Gambar 2. 13 Servo Dynamixel AX-12A	14
Gambar 2.14 Arduino IDE	15
Gambar 3.1 Diagram Alur Metode Penelitian	16
Gambar 3.2 Diagram Block Sistem	18
Gambar 5. 1 Topik Pembahasan Implementasi Teknik Enkoding Digital Pembacaan Sensor Ultrasonik Untuk Memetakan Keputusan Aksi Robot Quadruped	23
Gambar 5. 2 Desain Satu Kaki Robot	23
Gambar 5. 3 Desain Robot <i>Quadruped</i>	24
Gambar 5. 4 Skematik Rangkaian Sensor Ultrasonik HC-SR04 dan Arduino Mega	25
Gambar 5. 5 Diagram Alir Perancangan Perangkat Lunak Pengambilan Data Sensor	27
Gambar 5. 6 Diagram Alir Perancangan Teknik Enkoding Digital	28
Gambar 5. 7 Penamaan Sensor	29
Gambar 5. 8 Implementasi Robot <i>Quadruped</i>	33
Gambar 5. 9 Implementasi Delapan Sensor Ultrasonik HC-SR04	33
Gambar 5. 10 Implementasi Servo AX-12, OpenCM9.04 dan Ekspansi OpenCM485	34

Gambar 6. 1 Pengujian Menggunakan Alat Ukur.....	39
Gambar 6. 2 Tampilan Serial Monitor Pengujian Menggunakan Sensor.....	39
Gambar 6. 3 Tampilan Waktu Komputasi pada Serial Monitor	44
Gambar 6. 4 Grafik Pengujian Waktu Komputasi	46



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Sekarang ini perkembangan teknologi telah menciptakan berbagai kemajuan dibidang teknologi, terutama di bidang robotika. Beberapa tahun terakhir robot berkembang dengan menggantikan peran manusia terutama dalam lingkungan yang berbahaya, seperti daerah radiasi nuklir, penjelajahan ruang angkasa, penjinakan bom, dan lain-lain (Awal, 2015). Robot merupakan sebuah alat mekanik yang dapat melakukan tugas fisik, baik menggunakan kontrol manusia, atau memanfaatkan kecerdasan buatan. Robot memiliki kontruksi diantaranya *Robot Mobile* (Robot Bergerak), *Robot Manipulator*, *Robot humanoid*, *Flying Robot*, Robot Berkaki, Robot Jaringan, *Robot Animalia*, dan *Robot Cyborg*. Masing-masing kontruksi memiliki kontruksi dan mekanisme bergerak yang berbeda serta aplikasi yang beragam (Adriansyah, 2014).

Merancang sebuah robot bergerak, ada beberapa masalah yang harus diatasi. Salah satunya adalah masalah dalam merancang kemampuan robot untuk mengetahui posisi pada suatu waktu tertentu. Masalah tersebut disebut sistem lokalisasi robot. Terdapat tiga pendekatan dalam melakukan proses lokalisasi robot yaitu: *relative* atau *local localization*, *absolut* atau *global localization*, dan *probabilistic localization* (Adriansyah, 2014). Berdasarkan alat gerak robot, robot dapat diklasifikasikan menjadi 2 (dua) jenis yaitu robot berkaki dan robot beroda. Robot berkaki merupakan robot yang dapat bermanuver dengan kaki-kaki buatan. Robot berkaki memiliki beberapa keuntungan yaitu mampu bergerak di daerah yang halus atau kasar, bergelombang dan tidak rata atau datar, memanjat tangga, menghindar dan melangkah di atas halangan (Timur, Fanany, & Wajiansyah, 2017).

Kementrian Riset membuat sebuah wadah pengembangan robot di bidang pendidikan yaitu Kontes Robot Indonesia (KRI). Pada kontes tersebut salah satu kategorinya adalah Kontes Robot Pemadam Api (KRPAI). Pada Kontes Robot Pemadam Api (KRPAI), robot menggunakan alat gerak berupa kaki. Terdapat aturan yang harus dipenuhi untuk menyelesaikan tugas yang telah ditentukan. Sehingga, diperlukan gerakan robot yang efisien dan cepat agar robot bisa menyelesaikan tantangan dengan lebih tepat (KRI, 2017). Jenis Robot Pemadam Api berdasarkan jumlah kakinya dibagi menjadi dua jenis yaitu robot dengan enam kaki yang biasa disebut *Hexapod* dan robot dengan empat kaki atau yang biasa disebut *Quadruped*. Dari kedua desain tersebut, desain robot *Quadruped* lebih memungkinkan robot dapat bergerak kesegala arah tanpa harus memutar tubuh robot terlebih dahulu. Sehingga dengan desain *Quadruped* yang dapat berjalan segala arah membuat navigasi robot menjadi lebih efisien dan tidak membuang waktu untuk memutar tubuh robot.

Pada robot berkaki diharuskan mampu menghindari halangan dinding atau benda lainnya, dengan menggunakan sensor ultrasonik serta mencapai target berupa sebuah posisi. Sensor ultrasonik adalah sensor jarak yang menggunakan prinsip pantulan gelombang suara. Saat ini, banyak penelitian tentang kontrol pergerakan robot, diantaranya sejauh mana robot dapat menghindari halangan. Robot harus mampu mengetahui posisi halangan, sehingga mampu bergerak menghindari ke arah yang aman (Ningrum, Alasiry, Ardilla, & Kurnianto, 2010).

Pergerakan robot *quadruped* yang mampu bergerak ke 8 arah, membuat robot membutuhkan 8 sensor untuk memastikan area tersebut tidak terdapat halangan. 8 sensor tersebut memastikan bahwa area tersebut ada atau tidak halangan sehingga akan didapatkan 256 kemungkinan. 256 kemungkinan tersebut perlu di petakan menjadi 8 arah gerak robot, sehingga diperlukan teknik encoding digital yang merupakan rangkaian kombinasional yang menghasilkan kode spesifikasi pada keluarannya seperti biner sebagai respons terhadap satu atau lebih masukan.

Berdasarkan fakta yang telah disampaikan dibutuhkan sebuah robot yang memiliki navigasi segala arah yang efisien dan stabil dengan menerapkan teknik encoding digital yang akan dikembangkan dalam skripsi ini.

1.2 Rumusan Masalah

Berdasarkan uraian pada latar belakang tersebut, maka pada skripsi ini dapat dirumuskan beberapa permasalahan yaitu sebagai berikut :

1. Bagaimana tingkat keakuratan pembacaan sensor ultrasonik terhadap lingkungan sekitar?
2. Bagaimana penerapan teknik encoding digital pada robot *quadruped*?
3. Bagaimana tingkat akurasi pada sistem pengambilan keputusan robot?

1.3 Tujuan

Dari permasalahan tersebut adapun maksud dan tujuan dari penelitian ini adalah sebagai berikut :

1. Melakukan pengujian dan perhitungan terhadap tingkat keakuratan pembacaan sensor ultrasonik terhadap lingkungan sekitar.
2. Menerapkan teknik encoding digital pada robot *quadruped*.
3. Melakukan pengujian dan perhitungan tingkat akurasi pada sistem pengambilan keputusan robot.

1.4 Manfaat

Manfaat dari penelitian ini adalah :

1. Membuat sistem navigasi robot yang cepat dan efisien.
2. Menjadi alternatif pemodelan robot yang dapat bernavigasi dengan cepat dan stabil.

1.5 Batasan Masalah

Penelitian ini memiliki batasan masalah yang ditentukan agar permasalahan yang dirumuskan dapat lebih berfokus dan tidak meluas. Adapun batasan masalah pada penelitian ini antara lain:

1. Robot menggunakan sensor ultrasonik sejumlah delapan sensor sebagai sensor jarak.
2. Navigasi robot menggunakan teknik enkoding digital.
3. Robot memiliki 8 gerakan diantaranya maju, mundur, kanan, kiri, serong kanan depan, serong kiri depan, serong kanan belakang, dan serong kiri belakang.

1.6 Sistematika Pembahasan

Sistematik Penulisan sebagai berikut

Bab 1 Pendahuluan

Membahas mengenai latar belakang atau pokok permasalahan yang akan dibahas, menentukan rumusan dari masalah, tujuan, manfaat, dan metode yang dipakai untuk menyelesaikan masalah secara keseluruhan

Bab 2 Landasan Kepustakaan

Membahas tentang dasar teori yang digunakan sebagai acuan penulis mengerjakan sistem ini.

Bab 3 Metodologi

Membahas tentang metode, diagram alir pengerjaan, apa saja yang nanti dikerjakan dalam membuat sistem ini.

Bab 4 Rekayasa Kebutuhan

Membahas tentang gambaran umum sistem, analisis kebutuhan fungsional, kebutuhan perangkat keras, kebutuhan perangkat lunak dan batasan desain sistem.

Bab 5 Perancangan dan Implementasi

Bab ini membahas tentang perancangan sistem untuk implementasi teknik enkoding digital pada robot quadruped dan membahas mengenai proses implementasi.

Bab 6 Pengujian dan Analisis

Bab ini membahas mengenai cara pengujian dan analisis pada hasil pengujian sistem.

Bab 7 Kesimpulan dan Saran

Bab ini membahas tentang kesimpulan yang diperoleh dari perancangan, implementasi dan pengujian sistem, serta saran-saran yang didapatkan untuk pengembangan penelitian selanjutnya.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini berisikan tentang tinjauan pustaka yang menjabarkan mengenai kajian pustakan dan mengenai dasar teori yang diperlukan selama penelitian. Kajian pustakan membahas mengenai penelitian yang telah ada dan penelitian yang akan diusulkan. Sedangkan dasar teori membahas mengenai teori yang diperlukan selama menyusun penelitian yang diusulkan.

2.1 Tinjauan Pustaka

2.1.1 Desain dan Implementasi Kendali Cerdas untuk Robot *Quadpod* (Berkaki Empat) Studi Kasus Robot Pemadam Api

Penelitian awal di bidang robotika modern didukung oleh kepentingan-kepentingan komersial, sasarannya secara ekonomis adalah mengembangkan robot sebagai peningkatan produktivitas dan konsistensi. Salah satu pengembangan robot modern adalah robot pemadam api. Robot pemadam api tidak hanya sekedar memadamkan api melainkan mencari ruang yang terdapat api. Ketika robot mencari api yang terdapat didalam ruangan, robot melakukan proses berjalan dan mengetahui lingkungan sekitar agar dapat mengetahui posisi robot berada dan mengetahui tindakan apa yang harus di ambil, pada penentuan posisi tidak hanya mengetahui dimana robot berada tapi juga mengetahui posisi arah robot.

Robot ini memiliki tiga input yang diperoleh dari 3 sensor yaitu Sensor *Front*, sensor *left* dan sensor *right*. Ketiga input tersebut kemudian diproses menggunakan Fuzzy logic yang akan menghasilkan *output* berupa posisi robot. Kendali cerdas dalam robot tersebut disimulasikan melalui matlab. Proses yang dilakukan adalah pengambilan data menggunakan sensor. Setelah itu, ketiga data tersebut kemudian disimpan dalam variabel *F, R* dan *L*. *F, R*, dan *L* memiliki rentang nilai dari 5 sampai 30, memiliki 3 *Membership* yaitu *Low*, *med*, dan *high*. *Output* memiliki rentang nilai dari 0 sampai 4 yang terdiri dari 5 *Membership function* yaitu belok kanan, serong kanan, maju, serong kiri dan belok kiri (Timur, Fanany, & Wajiansyah, 2017).

2.1.1 Algoritma Fuzzy Logic dan *Wallfollower* pada Sistem Navigasi Robot *Hexapod* Berbasis Mikrokontroler AVR

Beberapa tahun terakhir ini perkembangan teknologi dibidang robotika telah menjadi perhatian yang cukup serius. Robot merupakan perangkat yang otomatis mampu bergerak secara sendiri untuk menyelesaikan suatu pekerjaan. Metode *wallfollower* adalah salah satu metode untuk navigasi robot, metode yang dipakai adalah menelusuri dinding kanan. Untuk menelusuri dinding kanan ini, digunakan tiga buah sensor SRF 04. Metode ini didukung dengan fuzzy logic yang berfungsi untuk menentukan langkah kaki robot yang harus diberikan kepada mikrokontroler. Mikrokontroler yang dipakai pada sistem robot ini adalah

ATMega128. Pada penerapan algoritma fuzzy logic dan metode *wallfollower* pada sistem navigasi robot *hexapod* berjalan dengan baik dalam arena yang telah dirancang (Awal, 2015).

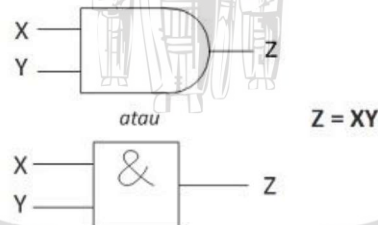
2.2 Dasar Teori

2.2.1 Gerbang Logika

Gerbang logika adalah dasar pembentuk Sistem Elektronika Digital yang berfungsi untuk mengubah satu atau beberapa masukan menjadi sebuah sinyal keluaran yang logis. Gerbang logika beroperasi berdasarkan sistem bilangan biner yang hanya memiliki 2 kode simbol yakni 1 dan 0 dengan menggunakan Teori Aljabar Boolean.

Terdapat tujuh jenis Gerbang Logika dasar yang membentuk sebuah Sistem Elektronika Digital, yaitu Gerbang *AND*, Gerbang *OR*, Gerbang *NOT*, Gerbang *NAND*, Gerbang *NOR*, Gerbang *X-OR*, dan Gerbang *X-NOR*. Tabel yang berisikan kombinasi-kombinasi Variabel masukan yang menghasilkan keluaran logis disebut dengan Tabel Kebenaran.

Gerbang *AND* memerlukan dua atau lebih masukan untuk menghasilkan hanya 1 keluaran. Simbol yang menandakan Operasi Gerbang Logika *AND* adalah tanda titik (.) atau tidak memakai tanda sama sekali, contohnya $Z=X.Y$ atau $Z=XY$. Gambar 2.1 merupakan simbol Gerbang *AND*. Tabel 2.1 merupakan tabel kebenaran Gerbang *AND* (Kho, 2015).



Gambar 2.1 Simbol Gerbang *AND*

Sumber : (Kho, 2015)

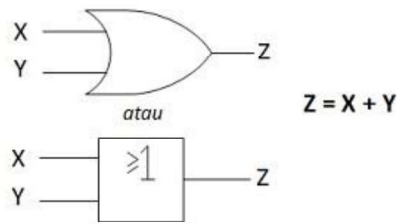
Tabel 2.1 Tabel Kebenaran Gerbang *AND*

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

Sumber : (Kho, 2015)

Gerbang *OR* memerlukan dua atau lebih masukan untuk menghasilkan hanya 1 keluaran. Simbol yang menandakan Operasi Gerbang Logika *OR* adalah

tanda plus (+), contohnya $Z=X+Y$. Gambar 2.2 merupakan simbol Gerbang *OR*. Tabel 2.2 merupakan tabel kebenaran Gerbang *OR*.



Gambar 2.2 Simbol Gerbang *OR*

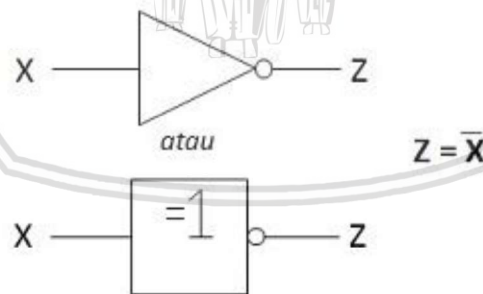
Sumber : (Kho, 2015)

Tabel 2.2 Tabel Kebenaran Gerbang *OR*

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

Sumber : (Kho, 2015)

Gerbang *NOT* memerlukan sebuah masukan untuk menghasilkan hanya 1 keluaran. Simbol yang menandakan Operasi Gerbang Logika *NOT* adalah tanda *minus* (-) di atas variabel inputnya atau tanda petik satu ('). Gambar 2.3 merupakan simbol Gerbang *NOT*. Tabel 2.3 merupakan tabel kebenaran Gerbang *NOT*.



Gambar 2.3 Simbol Gerbang *NOT*

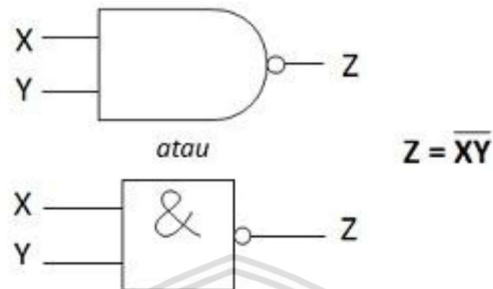
Sumber : (Kho, 2015)

Tabel 2.3 Tabel Kebenaran Gerbang *NOT*

X	Z
0	1
1	0

Sumber : (Kho, 2015)

Gerbang *NAND* adalah *NOT AND* atau BUKAN *AND*, gerbang *NAND* merupakan kombinasi dari gerbang *AND* dan gerbang *NOT* yang menghasilkan kebalikan dari keluaran Gerbang *AND*. Gambar 2.4 merupakan simbol Gerbang *NAND*. Tabel 2.4 merupakan tabel kebenaran Gerbang *NAND*.



Gambar 2.4 Simbol Gerbang *NAND*

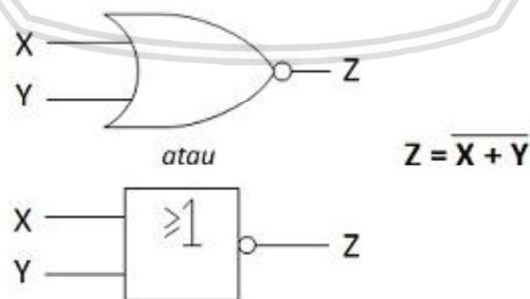
Sumber : (Kho, 2015)

Tabel 2.4 Tabel Kebenaran Gerbang *NAND*

X	Y	Z
0	0	1
0	1	1
1	0	1
1	1	0

Sumber : (Kho, 2015)

Gerbang *NOR* adalah *NOT OR* atau BUKAN *OR*, gerbang *NOR* merupakan kombinasi dari gerbang *OR* dan gerbang *NOT* yang menghasilkan kebalikan dari keluaran Gerbang *OR*. Gambar 2.5 merupakan simbol Gerbang *NOR*. Tabel 2.5 merupakan tabel kebenaran Gerbang *NOR*.



Gambar 2.5 Simbol Gerbang *NOR*

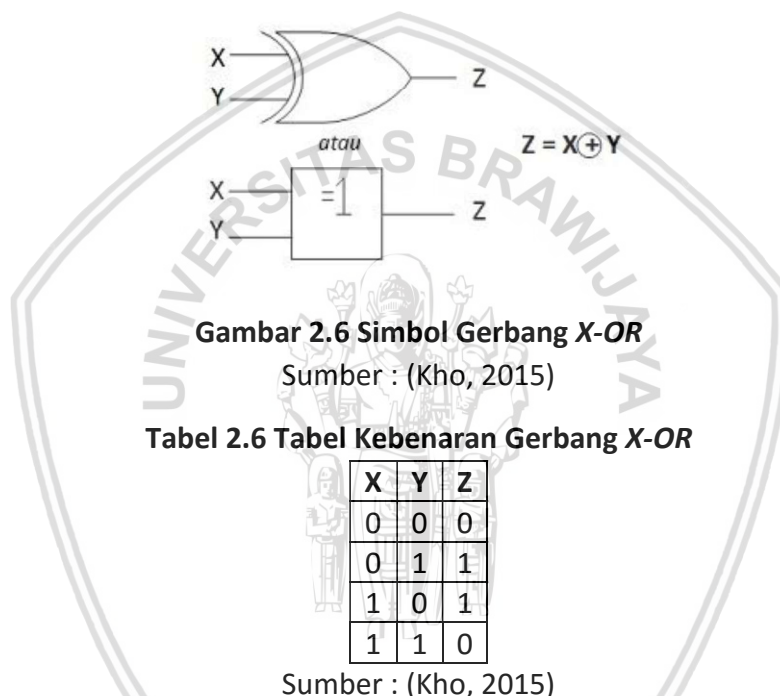
Sumber : (Kho, 2015)

Tabel 2.5 Tabel Kebenaran Gerbang NOR

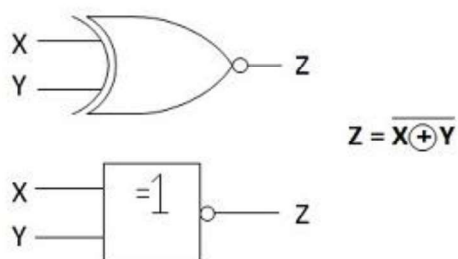
X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	0

Sumber : (Kho, 2015)

Gerbang X-OR memerlukan dua masukan untuk menghasilkan hanya 1 keluaran. Gambar 2.6 merupakan simbol Gerbang X-OR. Tabel 2.6 merupakan tabel kebenaran Gerbang X-OR.



Gerbang X-NOR memerlukan dua masukan untuk menghasilkan hanya 1 keluaran. Gambar 2.7 merupakan simbol Gerbang OR. Tabel 2.7 merupakan tabel kebenaran Gerbang OR.



Tabel 2.7 Tabel Kebenaran Gerbang X-NOR

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

Sumber : (Kho, 2015)

2.2.2 Teknik Enkoding Digital

Teknik Enkoding Digital adalah rangkaian kombinasional yang menghasilkan kode spesifik pada keluarannya seperti biner atau BCD sebagai respons terhadap satu atau lebih masukan aktif. Ada dua tipe teknik enkoding digital yaitu enkoding biner dan prioritas enkoding. Enkoding Biner mengubah salah satu dari 2^n input menjadi output n -bit. Kemudian Enkoding Biner memiliki *output* yang lebih sedikit dari pada kode *input*. Enkoding Biner berguna untuk mengompresi data dan dapat dibangun dari gerbang *AND* dan *OR* sederhana. Salah satu kelemahan utama dari Enkoding Biner standar adalah menghasilkan kesalahan pada keluarannya jika lebih dari satu masukan aktif pada waktu yang bersamaan. Untuk mengatasi masalah ini enkoding prioritas dikembangkan. Prioritas Enkoding adalah jenis lain dari rangkaian kombinasional mirip dengan Enkoding Biner, kecuali menghasilkan kode *output* berdasarkan *input* tertinggi yang diprioritaskan. Enkoding prioritas digunakan secara ekspansif adalah sistem digital dan komputer sebagai pengendali *interupsi mikroprosesor* di mana mereka mendeteksi *input* prioritas tertinggi (Electronics-Tutorials, 2017).

Sebagai contoh dua keluaran dengan empat masukan menggunakan teknik enkoding digital. Contoh tabel kebenaran dapat di lihat pada gambar 2.8

Gambar 2. 8 Contoh Tabel Kebenaran Empat Masukan Dua Keluaran

Q ₀	Q ₁	Q ₂	Q ₃	A	B
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

Sesuai dengan tabel kebenaran di atas maka didapat dua persamaan sebagai keluaran yaitu :

$$A = Q_0' Q_1' Q_2 Q_3' + Q_0' Q_1' Q_2' Q_3$$

$$B = Q_0' Q_1 Q_2' Q_3' + Q_0' Q_1' Q_2' Q_3$$

2.2.3 Robot

Robot adalah mesin yang dirancang untuk menjalankan satu atau lebih tugas secara otomatis dengan kecepatan dan presisi (Rouse, 2016). Berdasarkan alat geraknya robot diklasifikasikan menjadi 2 (dua) jenis yaitu robot beroda dan robot berkaki.

Tabel 2.8 Spesifikasi Arduino Mega 2560

Microcontroller	Atmega2560
Operating Voltage	5V
Input Voltage (Recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (15 dapat digunakan provide PWM output)
PWM Digital I/O Pins	15
Analog Input Pins	16
DC Current per I/O	20 mA
DC Current for 3.3V pin	50 mA
Flash Memory	256KB dengan 8 KB digunakan bootloader
SRAM	9 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED-BUILTIN	13
Length	101.52 mm
Width	53.3 mm
Weigth	37 g

Sumber : (Arduino, 2018)

2.2.5 Sensor Ultrasonik HC-SR04

Sensor ultrasonik adalah sensor yang bekerja menggunakan prinsip pantulan gelombang suara, dimana sensor ini mengeluarkan gelombang suara yang kemudian ditangkapnya kembali dengan berdasarkan perbedaan waktu sebagai pengindraannya. Perbedaan waktu yang dihasilkan antara gelombang suara yang dipancarkan dengan gelombang suara yang ditangkapnya kembali gelombang tersebut adalah berbanding lurus dengan jarak atau tinggi objek yang memantulkannya. Jenis objek-objek yang dapat diindera yaitu: objek benda padat, benda cair, dan butiran maupun tekstil.



Gambar 2.10 HC-SR04

Sumber : (Arduinolearning, 2017)

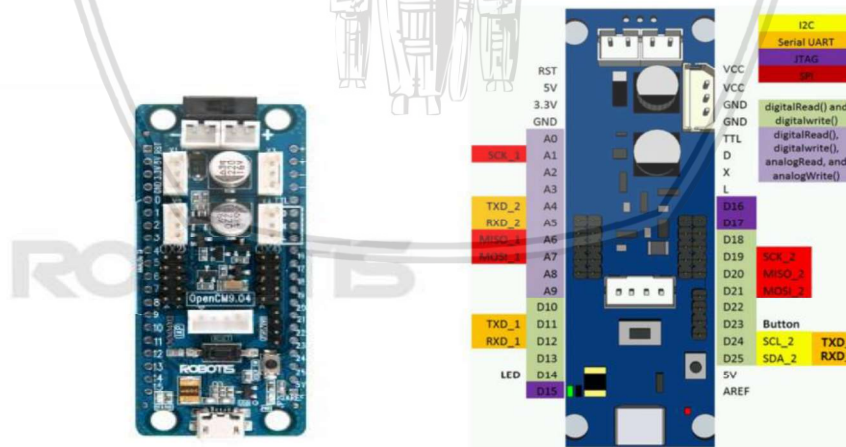
Gambar 2.10 merupakan sensor ultrasonik HC-SR04. HC-SR04 merupakan salah satu modul sensor jarak ultrasonik dengan jarak yang dapat diukur 3 - 400 cm. Pengukuran jarak modul tersebut memiliki akurasi mencapai 3 mm. Pada modul sensor tersebut terdapat ultrasonik yang berfungsi sebagai *transmitter*, ultrasonik yang berfungsi sebagai *receiver* dan *control sirkuit*. Cara kerja modul sensor ultrasonik HC-SR04 yaitu, menggunakan *Input Output* untuk memicu setidaknya 10us high level sinyal, modul tersebut secara otomatis akan mengirimkan delapan 40 kHz dan akan mendeteksi ada atau tidak sinyal *pulse* yang kembali, jika sinyal tersebut kembali melalui *high level* waktu durasi *high output* *Input Output* adalah waktu dari mengirim gelombang ultrasonik sampai gelombang tersebut kembali, test jarak = $(\text{high level time} \times \text{velocity of sound (340 M/S)} / 2)$ (Elecbreaks Team, 2015). Pada Tabel 2.9 menjelaskan parameter HC-SR04.

Tabel 2.9 Parameter Sensor Ultrasonik HC-SR04

Working Voltage	DC 5 V
Working Current	15Ma
Working Frequency	40Hz
Max Range	4m
Min Range	3cm
Measuring Angle	15 degree

Sumber : (Team, 2015)

2.2.6 OpenCM9.04



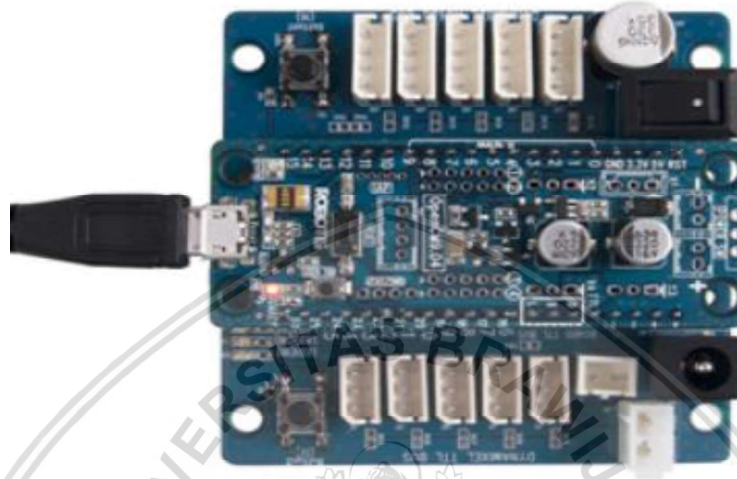
Gambar 2.11 Layout dan Pin Mapping OpenCM9.04

Sumber : (ROBOTIS, OpenCM9.04, 2010)

OpenCM9.04 merupakan papan mikrokontroler yang memiliki basis pada 32bit ARM cortex-M3 dengan *layout* seperti Gambar 2.11. Skema pada OpenCM9.04 dan kode sumber memiliki sifat *open-source*. Tersedia 3 tipe dan masing-masing tipe memiliki perbedaan pada tersedianya konektor. OpenCM9.04

mempunyai beberapa kelebihan diantaranya telah terdapat pin analog didalamnya, dan dapat digunakan untuk mengontrol servo dinamiksel AX-12. Gambar 2.11 merupakan gambar pin yang terdapat pada OpenCM9.04 (ROBOTIS, OpenCM9.04, 2010).

2.2.7 Board Ekspansi OpenCM485



Gambar 2.12 Layout OpenCM485 dengan OpenCM9.04

Sumber : (ROBOTIS, OPENCM 485 EXPANSION BOARD, 2017)

Ekspansi OpenCM 485 yang terdapat pada Gambar 2.13 merupakan *board ekspansi* sebagai pengontrol servo dinamiksel. *Ekspansi* OpenCM 485 tersebut terhubung kepada pengendali OpenCM9.04. Papan pengontrol OpenCM9.04 diperlukan untuk digunakan OpenCM 485 *Expansion Board*. Kontroler OpenCM9.04 sebagai pendukung Dynamixel menggunakan RS-485 dan TTL (ROBOTIS, OPENCM 485 EXPANSION BOARD, 2017).

2.2.8 Servo Dynamixel AX-12A

Salah satu aktuator, yaitu servo AX-12A dari Robotis merupakan aktuator paling baik yang saat ini ada di pasaran dan telah menjadi standar defacto sebagai generasi berikutnya pada bidang robotika. Servo AX-12A pada Gambar 2.13 memiliki sebuah kemampuan untuk dapat melacak kecepatan, suhu, posisi poros, tegangan, dan beban. Algoritma kontrol yang digunakan untuk mempertahankan posisi poros pada aktuator ax-12 tidak cukup untuk dapat disesuaikan secara individual pada masing-masing servo, yang memungkinkan untuk dapat mengendalikan kecepatan dan mengendalikan kekuatan respons motor. Seluruh manajemen sensor dan pengontrol posisi dilakukan oleh mikrokontroler servo *built-in*. Pendekatan terdistribusi tersebut membuat pengendali utama bebas untuk melakukan fungsi lainnya (Robotis, 2016).



Gambar 2. 13 Servo Dynamixel AX-12A

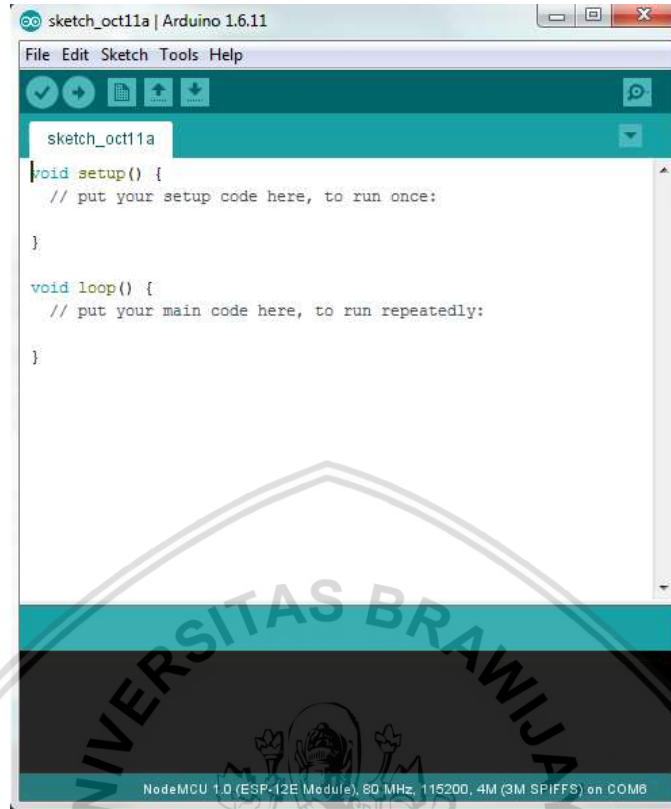
Sumber : (Robotics, 2018)

2.2.9 Arduino IDE

Arduino adalah pengendali *mikro single-board* yang bersifat *open-source*, diturunkan dari Wiring Platform, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardwarenya* memiliki prosesor Atmel AVR dan *softwarenya* memiliki bahasa pemrograman sendiri.

Arduino juga merupakan platform *hardware* terbuka yang ditujukan kepada siapa saja yang ingin membuat purwarupa peralatan elektronik interaktif berdasarkan *hardware* dan *software* yang fleksibel dan mudah digunakan. Mikrokontroler diprogram menggunakan bahasa pemrograman arduino yang memiliki kemiripan *syntax* dengan bahasa pemrograman C. Karena sifatnya yang terbuka maka siapa saja dapat mengunduh skema *hardware* arduino dan membangunnya.

Arduino menggunakan keluarga mikrokontroler ATmega yang dirilis oleh Atmel sebagai basis, namun ada individu/perusahaan yang membuat *clone* arduino dengan menggunakan mikrokontroler lain dan tetap kompatibel dengan arduino pada level *hardware*. Untuk fleksibilitas, program dimasukkan melalui *bootloader* meskipun ada opsi untuk *bypass bootloader* dan menggunakan *downloader* untuk memprogram mikrokontroler secara langsung melalui port ISP. Pada Gambar 2.14 adalah tampilan Arduino IDE.



Gambar 2.14 Arduino IDE

2.2.10 FreeRTOS

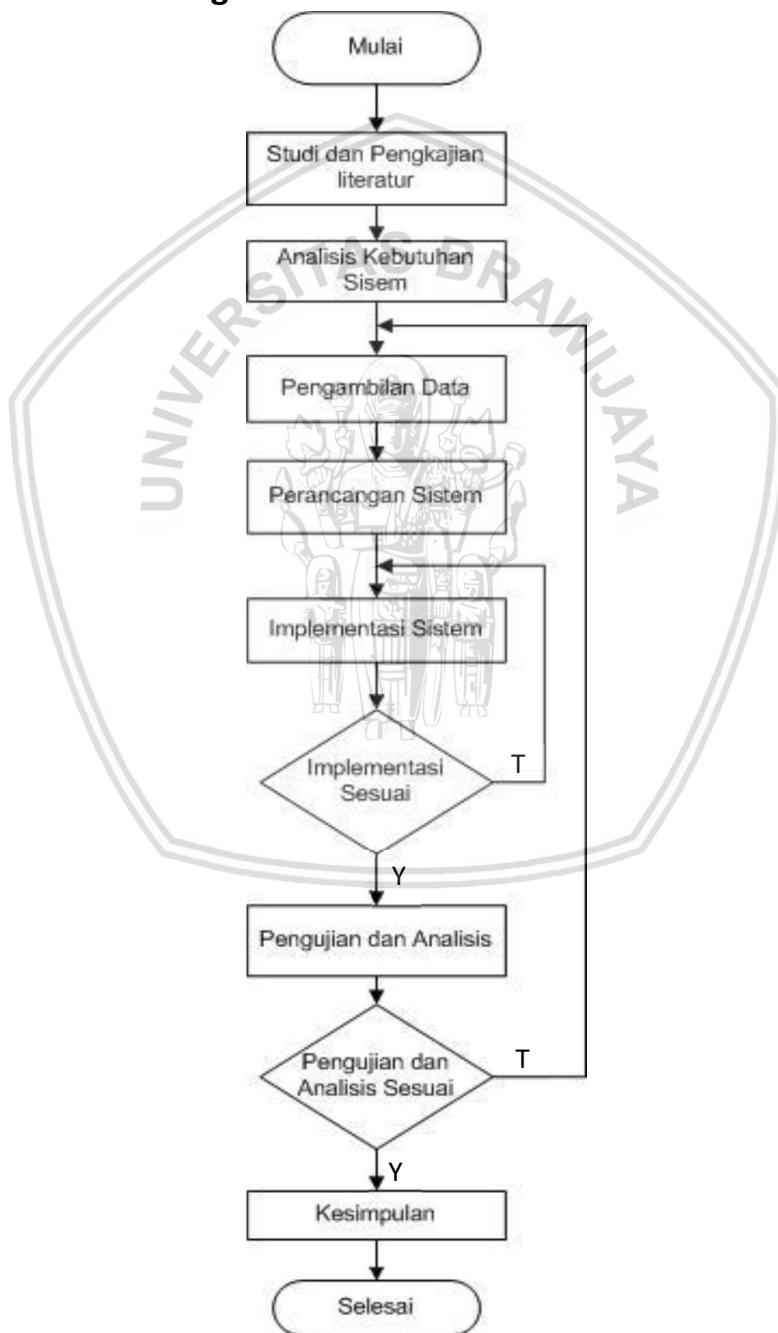
FreeRTOS yang disediakan oleh *Real Time Engineers Ltd.*, adalah sistem operasi *real-time* yang bersifat *open source*, terukur, dan terdokumentasi dengan baik, yang telah didukung lebih dari 30 arsitektur perangkat keras berbeda. Semua fitur membuat *FreeRTOS* menjadi sistem operasi *real time* yang populer. Selain itu, sumber-sumbernya sebagian besar ditulis dalam Bahasa pemrograman C dengan sangat sedikit rutinitas perakitan ekstra, yang membuat sistem portabel dan mudah dipahami (Saputro, 2018).

Fitur lain yang terlihat adalah ukuran kernelnya yang kecil: membutuhkan sekitar 5 hingga 10 kB ROM, melihat pada kompilator, arsitektur dan konfigurasi kernel. Sisi lain dibutuhkan dari hingga beberapa kB RAM untuk semua RAM yang tersedia, tergantung pada jumlah tugas, jumlah antrian yang dibuat dan skema alokasi memori. Setiap tugas memiliki tumpukan yang dialokasikan sendiri (ukuran tumpukan *default* terkecil menjadi 140 kata untuk implementasi ARM Cortex M3. *FreeRTOS* juga menyediakan 3 mode untuk mengimplementasikan *heap*. Mode yang paling sederhana mengimplementasikan alokasi memori, tetapi tidak ada dealokasi memori atau penggunaan kembali memori sementara mode yang paling kompleks mengimplementasikan baik alokasi dan pengalokasian (Saputro, 2018).

BAB 3 METODOLOGI

Pada bab Metodologi ini menjelaskan mengenai langkah-langkah yang akan dilakukan dalam menyusun skripsi, yang meliputi studi dan pengkajian literatur, analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian dan analisis, sehingga dapat ditarik kesimpulan.

3.1 Alur Metodologi Penelitian



Gambar 3.1 Diagram Alur Metode Penelitian

Sesuai dengan Gambar 3.1, alur penelitian ini diawali dengan studi literatur yaitu mencari literatur pendukung yang berfungsi sebagai bahan acuan melakukan penelitian. Selanjutnya, analisis kebutuhan dan melakukan pengambilan data sesuai dengan studi literatur yang di dapat. Setelah pengambilan data selesai dilakukan, implementasi sistem dengan melakukan perancangan terlebih dahulu. Untuk melakukan pengujian dan analisis ada tahapan yang memiliki syarat untuk dapat dilanjutkan ke tahapan selanjutnya, jika implementasi tersebut sesuai dengan keseluruhan yang diirancang sebelumnya, maka sistem dapat dinyatakan siap untuk melakukan pengujian dan analisis. Begitu pula dengan pengujian dan analisis, ketika pengujian dan analisis telah sesuai dengan hipotesis awal, sehingga dapat ditarik kesimpulan dari keseluruhan tahapan yang telah dilakukan. Jika tidak sesuai, maka tahapan akan kembali ke perancangan sistem.

3.2 Studi Literatur

Untuk melakukan perancangan dan implementasi penelitian ini, perlu dilakukan studi literatur. Studi literatur memiliki tujuan untuk mengkaji hal-hal yang memiliki hubungan dengan teori-teori yang dapat mendukung dalam perencanaan yang akan dibuat. Studi literatur juga digunakan sebagai teori penguat dan landasan dasar dalam penelitian. Teori pendukung tersebut bisa didapatkan dari buku, jurnal, *paper* dan internet. Adapun beberapa literatur yang digunakan pada penelitian ini meliputi:

1. Kontruksi robot berkaki 4 (quadruped).
2. Pembacaan sensor jarak ultrasonik dan mikrokontroler
3. Gerbang Logika dan Teknik Enkoding Digital

3.3 Analisis Kebutuhan

Analisis kebutuhan sistem memiliki tujuan untuk menganalisis keseluruhan kebutuhan yang dibutuhkan oleh sistem yang akan dibuat. Analisis kebutuhan sistem ini dilakukan dengan mengidentifikasi kebutuhan sistem tersebut. Analisis kebutuhan tersebut, dapat meliputi kebutuhan perangkat lunak dan kebutuhan perangkat keras. Perangkat lunak yang dibutuhkan dalam membangun sistem ini memiliki spesifikasi sebagai berikut:

1. Perangkat lunak yang dapat digunakan sebagai pembuat program dan dapat di-*upload* ke Mikrokontroler Arduino Mega yang digunakan.
2. Perangkat lunak yang digunakan dapat membuat program sehingga menggerakkan penggerak servo.

Sedangkan untuk perangkat keras yang dibutuhkan dalam membuat sistem ini memiliki spesifikasi sebagai berikut:

1. Perangkat keras yang bisa di-*upload* suatu program sehingga dapat membaca nilai sensor yang masuk, kemudian dapat mengirimkan data tersebut untuk dapat dilakukan proses perhitungan.

2. Perangkat keras yang mampu membaca kemudian menjadi input pada Mikrokontroler yang digunakan.
3. Perangkat keras yang mampu mengontrol alat gerak berupa servo dinamixel AX-12

3.4 Pengambilan Data

Pengambilan data ini dapat dilakukan dengan melakukan pengambilan nilai pada sensor. Seluruh data adalah data yang berbeda satu data dengan data yang lain. Data-data tersebut merupakan nilai 8 sensor ultrasonik. Data tersebut kemudian diolah menggunakan teknik encoding digital untuk mendapatkan keputusan robot bergerak ke arah mana.

3.5 Perancangan Sistem

Setelah semua kebutuhan telah didapatkan melalui tahap analisis kebutuhan dilakukan perancangan sistem. Perancangan sistem dibuat untuk menghubungkan perangkat lunak dan perangkat keras. Sistem tersebut, dapat dilihat gambaran umum sistem seperti pada Gambar 3.2.



Gambar 3.2 Diagram Block Sistem

Pada bagian input alat terdapat sensor ultrasonik sebagai pendeteksi jarak dinding atau lingkungan. Nilai 8 sensor tersebut akan dibaca oleh mikrokontroler, kemudian nilai sensor akan diolah. Sehingga di dapatkan keputusan berupa arah gerak robot.

3.6 Implementasi Sistem

Setelah perancangan sistem masuk ketahap berikutnya berupa implementasi sistem. Pada implementasi diawali dengan pengujian terhadap sensor yang di gunakan, dalam hal ini sensor ultrasonik. Nilai yang di dapat dari 8 sensor tersebut, akan dijadikan dasar nilai untuk mengambil keputusan arah gerak robot. Sehingga perangkat keras dan perangkat lunak diatur dan dibuat sesuai dengan perancangan yang telah di rancang sebelumnya. Selain itu, implementasi sistem harus dikerjakan secara bersamaan agar mampu membentuk sebuah sistem yang sesuai dengan rancangan.

3.7 Pengujian dan Analisis

Pada tahap pengujian ini dilakukan jika tahap implementasi sistem telah sesuai dengan tahapan pada perancangan sistem. Terdapat beberapa skenario yang akan dilakukan untuk menguji sistem, yaitu:

1. Pengujian sensor ultrasonik HC-SR04 terhadap cara kerja sensor sesuai dengan spesifikasi.
2. Pengujian waktu komputasi teknik encoding digital,
3. Pengujian akurasi hasil klasifikasi teknik encoding digital.

Pada tahap ini juga sebagai penentu penelitian dapat dilanjutkan ke tahap kesimpulan atau harus kembali ke tahap pengambilan data.

3.8 Kesimpulan

Kesimpulan di ambil jika semua tahapan metodologi penitian di lakukan. Mulai dari Studi dan penganalisis kebutuhan hingga pengujian dan analisis. Kesimpulan di ambil berdasarkan hasil akhir implementasi. Kesesuaian hasil akhir pada tujuan dari implementasi ini di jelaskan pada kesimpulan. Selanjutnya, pada kesimpulan juga berisi saran untuk mengembangkan dan memperbaiki penelitian ini. Saran tersebut bertujuan untuk perbaikan untuk penelitian selanjutnya.



BAB 4 REKAYASA KEBUTUHAN

Pada bab rekayasa kebutuhan ini dijelaskan secara rinci mengenai gambaran umum sistem, analisis kebutuhan fungsional, kebutuhan perangkat keras, kebutuhan perangkat lunak dan batasan desain sistem.

4.1 Gambaran Umum Sistem

Implementasi Teknik Enkoding Digital Pembacaan Sensor Ultrasonik untuk Memetakan Aksi Robot *Quadruped* merupakan suatu robot berkaki empat yang dapat mendeteksi halangan dan menjauhi halangan dengan menggunakan sensor ultrasonik. Adapun halangan yang di sekitar robot akan di deteksi dengan sensor ultrasonik HC-SR04 sebanyak delapan sensor. Berdasarkan nilai kedelapan sensor tersebut akan di dapatkan 256 kemungkinan yang mungkin terjadi. 256 kemungkinan itu akan menentukan delapan pergerakan yang dapat dilakukan sensor tersebut. Penggunaan teknik enkoding pada robot ini dikarenakan dalam melakukan klasifikasi, sudah diketahui terdapat delapan gerakan dari 256 kemungkinan. Selain itu enkoding digital adalah rangkaian kombinasional yang menghasilkan kode spesifik pada keluarannya seperti biner atau BCD sebagai respons terhadap satu atau lebih masukan aktif. Sehingga dapat menjadi tepat digunakan karna dari 256 kemungkinan tersebut akan memiliki output sebanyak 8 gerakan robot. Hasil dari perhitungan teknik enkoding digital akan otomatis menggerakkan robot menjauhi halangan di sekitarnya.

4.2 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan sebagai mengumpulkan semua kebutuhan yang diperlukan yang diperlukan untuk robot *quadruped* ini. Dalam melakukan analisis kebutuhan pada sistem ini terdiri dari beberapa kebutuhan yang perlu dijabarkan yakni meliputi kebutuhan fungsional dan kebutuhan non fungsional, kebutuhan non fungsional terdiri dari kebutuhan perangkat keras dan kebutuhan perangkat lunak.

4.2.1 Kebutuhan Fungsional

Berikut merupakan kebutuhan fungsional yang harus dapat dilakukan oleh sistem:

1. Sensor ultrasonik dapat membaca jarak halangan di sekitar robot yang dideteksi.
Sensor ultrasonik bertugas untuk membaca dan mendeteksi halangan disekitar. Jumlah sensor yang digunakan sebanyak delapan. Nilai jarak yang dibaca oleh sensor ini berupa nilai frekuensi. Nilai sensor yang di dapatkan berupa nilai digital yaitu 1/0. Nilai 1 memiliki jarak kurang dari 10 cm, dan

nilai 0 memiliki jarak lebih dari sama dengan 10 cm. Sehingga didapat jarak antara robot dengan halangan.

2. Sistem dapat melakukan perhitung menggunakan teknik encoding digital dari nilai masing-masing delapan sensor.
Pada fungsionalitas ini bertugas untuk melakukan proses penentuan gerakan robot menggunakan teknik encoding digital berdasarkan input dari delapan sensor.
3. Robot dapat bergerak sesuai hasil teknik encoding digital.
Pada fungsionalitas ini bertugas untuk menggerakkan robot dari sistem yaitu hasil teknik encoding digital, sehingga pergerakan robot akan menjauhi halangan di sekitar robot.
4. Sistem dapat berkomunikasi antara Arduino Mega dengan Open CM9.04
Pada fungsionalitas ini berfungsi sebagai mengirim data yang telah dikomputasi oleh Arduino Mega ke Open CM9.04 dengan menggunakan komunikasi serial.

4.2.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional pada sistem ini terdiri dari kebutuhan perangkat keras dan kebutuhan perangkat lunak yang dijelaskan secara rinci di bawah ini.

4.2.2.1 Kebutuhan Perangkat Keras

Sebagai pendukung implementasi pembuatan sistem pada sisi perangkat keras sehingga diperlukan beberapa alat, yang di jelaskan sebagai berikut:

1. Mikrokontroler Arduino Mega
Mikrokontroler Arduino digunakan sebagai otak dalam pengolahan informasi dalam perancangan sistem.
2. Sensor Ultrasonik HC-SR04
Sensor ultrasonik HC-SR04 dibutuhkan sebagai bagian dalam mendeteksi halangan sekitar robot.
3. OpenCM9.04
OpenCM9.04 diperlukan untuk menggunakan OpenCM 485 *Expansion Board*, sehingga dapat mengontrol servo dinamiksel.
4. Board Ekspansi OpenCM485
Board Ekspansi OpenCM489 diperlukan untuk board ekspansi untuk mengontrol servo dinamiksel.
5. Servo Dinamixel
Servo Dinamixel digunakan sebagai akuator untuk menggerakkan robot.
6. Kabel Jumper
Kabel jumper pada sistem tersebut dibutuhkan sebagai penghubung antara perangkat satu dengan perangkat lainnya.
7. Laptop

Laptop dalam hal ini berguna sebagai media dalam membuat program untuk mikrokontroler. Adapun Spesifikasi Laptop yang digunakan antara lain :

- Model Perangkat : HP Pavilion 14 Notebook PC
- Prosesor : Intel Core i5-5200U CPU @2.20GHz (4 CPUs)
- Sistem Operasi : Window 10 Pro 64-bit

4.2.2.1 Kebutuhan Perangkat Lunak

4. Pada kebutuhan perangkat lunak ini terdapat hal-hal yang dibutuhkan oleh sistem pada penelitian tersebut yaitu Arduino IDE yang dibutuhkan sebagai pengoperasikan mikrokontroler Arduino Mega dan sebagai penjalankan program lainnya yang dibutuhkan dalam pembuatan program dalam penelitian ini. Selain itu pada Arduino IDE ini terdapat fitur *library* yang berguna dalam membuat program yang mudah. *Library* yang digunakan adalah `#include <NewPing.h>` sebagai pembaca sensor dan `#include <Arduino_FreeRTOS.h>` mengimplementasikan RTOS.

4.3 Batasan Desain Sistem

Dalam pembuatan Implementasi Teknik Enkoding Digital Pembacaan Sensor Ultrasonik untuk Memetakan Keputusan Aksi Robot *Quadruped* ada beberapa batasan sehingga lingkup dalam pembahasan, perancangan, maupun pengimplemntasiannya tidak meluas. Adapun batasan-batasan desain sistem ini adalah sebagai berikut :

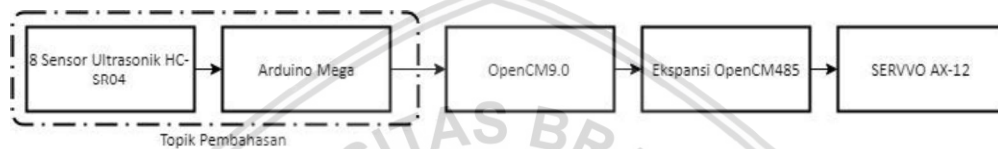
1. Robot menggunakan sensor ultrasonik sebagai sensor jarak sejumlah delapan.
2. Navigasi robot menggunakan algoritma teknik enkoding digital.
3. Robot memiliki 8 gerakan diantaranya maju, mundur, kanan, kiri, serong kanan depan, serong kiri depan, serong kanan belakang, dan serong kiri belakang.
4. Robot hanya dapat menjauhi halangan di sekitarnya hingga disekitarnya tidak terdapat halangan
5. Ukuran halangan yang dapat di deteksi memiliki panjang 8 cm dan tinggi 24 cm.

BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab 5 ini memaparkan secara terperinci baik perancangan dan implementasi pada sistem mengenai proses perancangan dan implementasi dari perangkat keras maupun dari perangkat lunak.

5.1 Perancangan Sistem

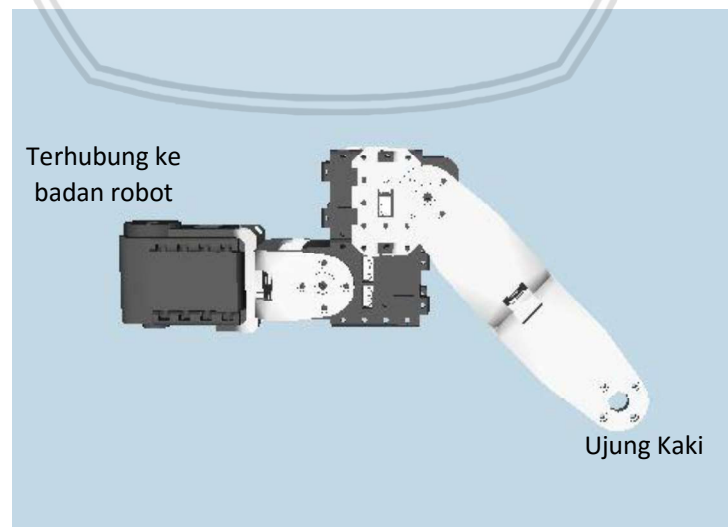
Pada perancangan sistem ini akan menjabarkan cara merancang sistem yang dimulai dari perancangan Robot Quadruped, perancangan perangkat keras, dan perancangan perangkat lunak sesuai dengan topik pembahasan pada gambar 5.1.



**Gambar 5. 1 Topik Pembahasan Implementasi Teknik Enkoding Digital
Pembacaan Sensor Ultrasonik Untuk Memetakan Keputusan Aksi Robot
Quadruped**

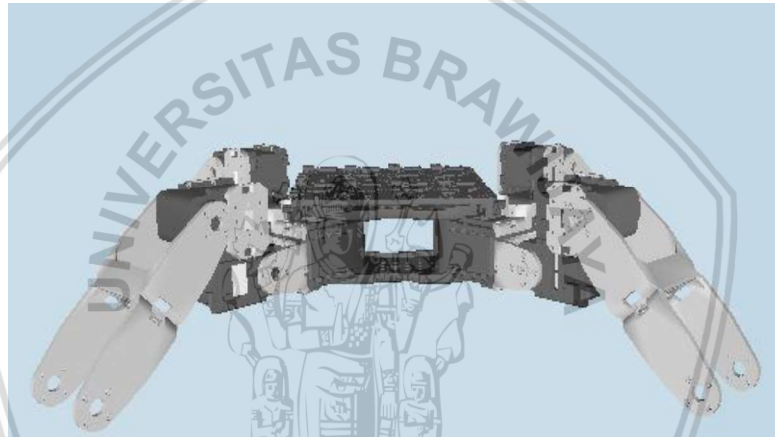
5.1.1 Perancangan Robot Quadruped

Dalam perancangan desain robot implementasi teknik enkoding digital tersebut perlu diperhatikan tentang peletakkan tiap-tiap bagian komponen serta ukuran alat yang akan dikembangkan atau digunakan. Pembuatan desain robot untuk menggambarkan bentuk robot yakni robot berkaki empat dengan delapan sensor ultrasonik HC-SR04. Bentuk desain satu kaki robot ditunjukkan pada Gambar 5.2 dan desain satu kaki robot tersebut dirangkai menjadi desain empat kaki robot seperti Gambar 5.3.



Gambar 5. 2 Desain Satu Kaki Robot

Posisi sensor ultrasonik HC-SR04 diletakkan mengarah ke sekitar robot untuk mendeteksi benda di sekitar robot. Sensor ultrasonik HC-SR04 tersebut terletak di bagian atas robot, sehingga halangan setinggi robot tersebut dapat terdeteksi. Peletakkan sensor ultrasonik HC-SR04 dibuat demikian agar dapat membaca jarak halangan dan menjauhinya. Halangan yang dapat dibaca oleh robot tersebut harus setinggi robot, dikarenakan sensor ultrasonik HC-SR04 yang hanya dapat membaca sensor lurus kedepan, sehingga sensor tidak dapat mendeteksi halangan apabila halangan lebih rendah dari pada tinggi robot tersebut. Nilai yang di dapatkan dari sensor ultrasonik HC-SR04 akan di proses oleh Arduino Mega untuk mendapatkan keputusan gerakan yang akan dilakukan oleh robot. Keputusan gerakan yang di dapatkan tersebut selanjutnya akan di kirim ke openCM9.04 yang digunakan sebagai *servo controller*.



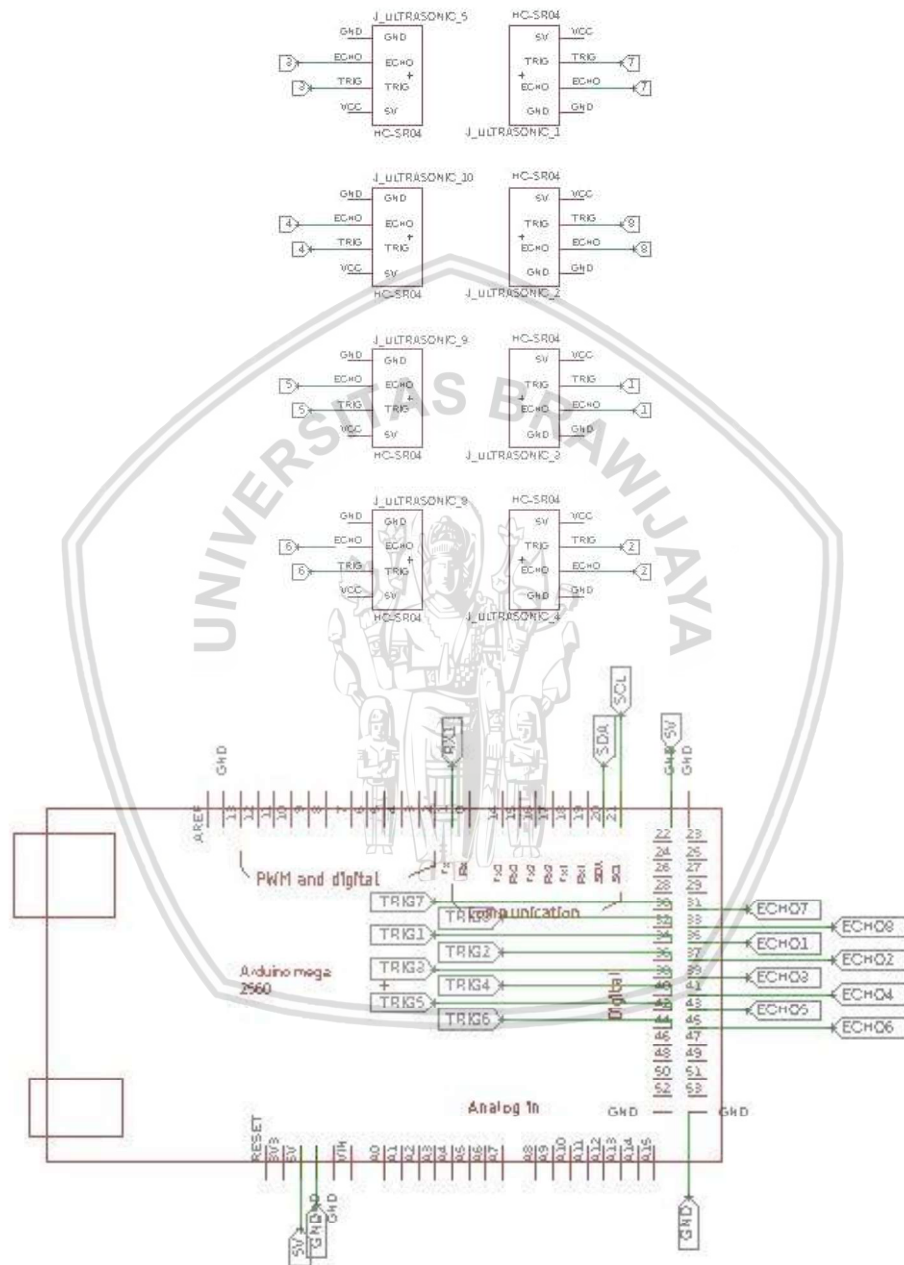
Gambar 5. 3 Desain Robot *Quadruped*

Sesuai dengan Gambar 5.3, robot tersebut memiliki empat kaki yang masing-masing kaki digerakan menggunakan servo AX-12. Dengan keempat kaki tersebut dapat melakukan delapan gerakan. Pergerakan yang dapat di lakukan oleh robot yaitu depan, belakang, kanan, kiri, serong kanan depan, serong kiri depan, serong kanan belakang, dan serong kiri belakang. Pergerakan robot akan menjauhi halangan ditentukan dengan menggunakan penerapan teknik enkoding digital.

5.1.2 Perancangan Perangkat Keras

Pada perancangan perangkat keras ini dilakukan perancangan berdasarkan analisis kebutuhan perangkat keras beserta spesifikasi dari masing-masing perangkat keras tersebut sehingga dapat membangun sistem sesuai dengan yang diharapkan. Pada tahap ini perancangan perangkat keras menjabarkan secara rinci mengenai hubungan skematik pin-pin yang digunakan antara tiap komponen-komponen pada perangkat keras, dalam hal ini sensor ultrasonik HC-SR04 sebagai *input*, dihubungkan dengan mikrokontroler arduino mega yang berfungsi sebagai pengolah data dan hasil olahan datanya akan dikirimkan ke OpenCM9.0 sehingga Ekspansi OpenCM485 dapat menggerakan servo AX-12 sesuai keinginan. Skematik

diagram perancangan perangkat keras Implementasi Teknik Enkoding Digital Pembacaan Sensor Ultrasonik untuk Memetakan Keputusan Aksi Robot *Quadruped* ini ditunjukkan pada Gambar 5.4.



Gambar 5. 4 Skematik Rangkaian Sensor Ultrasonik HC-SR04 dan Arduino Mega

Pada diagram skematik di atas terlihat bahwa ada Sensor Ultrasonik HC-SR04 yang memiliki fungsi mendeteksi halangan di sekitar robot. Sensor Ultrasonik HC-SR04 dihubungkan ke arduino mega dengan koneksi antar pin.

Tabel 5. 1 Tabel Pin Skematik

Sensor Ultrasonik HC-SR04 ke-	Pin Sensor Ultrasonik HC-SR04 ke Arduino Mega			
	Vcc	Trig	Echo	GND
1	5v	D30	D31	GND
2	5v	D32	D33	GND
3	5v	D34	D35	GND
4	5v	D36	D37	GND
5	5v	D38	D39	GND
6	5v	D40	D41	GND
7	5v	D42	D43	GND
8	5v	D44	D45	GND

Berdasarkan Tabel 5.1 tersebut terlihat bahwa delapan sensor Ultrasonik HC-SR04 dengan empat pin pada masing-masing sensor disambungkan secara langsung ke delapan belas pin Arduino Mega yaitu 5v, GND, D30, D32, D34, D36, D38, D40, D42, D44, D31, D33, D35, D37, D39, D41, D43, dan D45. Untuk pin VCC pada delapan sensor Ultrasonik HC-SR04 di hubungkan ke Pin 5V pada Arduino Mega. Untuk pin GND pada masing-masing sensor Ultrasonik HC-SR04 dihubungkan ke pin GND pada Arduino Mega. Selanjutnya untuk Pin Trig pada setiap sensor Ultrasonik HC-SR04 dihubungkan ke pin D30, D32, D34, D36, D38, D40, D42, dan D44 pada Arduino Mega. Sedangkan untuk Pin Echo pada masing-masing sensor ultrasonik dihubungkan ke pin D31, D33, D35, D37, D39, D41, D43, dan D45 pada Arduino Mega.

Komunikasi Arduino mega dan open CM9.04 dilakukan dengan menghubungkan pin Tx pada Arduino mega dan pin Rx pada OpenCM9.04 sehingga terjadi komunikasi secara serial. OpenCM9.04 menggunakan sumber tegangan dari baterai sebesar 12v. Selanjutnya OpenCM9.04 mengirimkan data ke ekspansi OpenCM485 sehingga Servo AX-12 dapat bergerak sesuai dengan gerakan yang diinginkan.

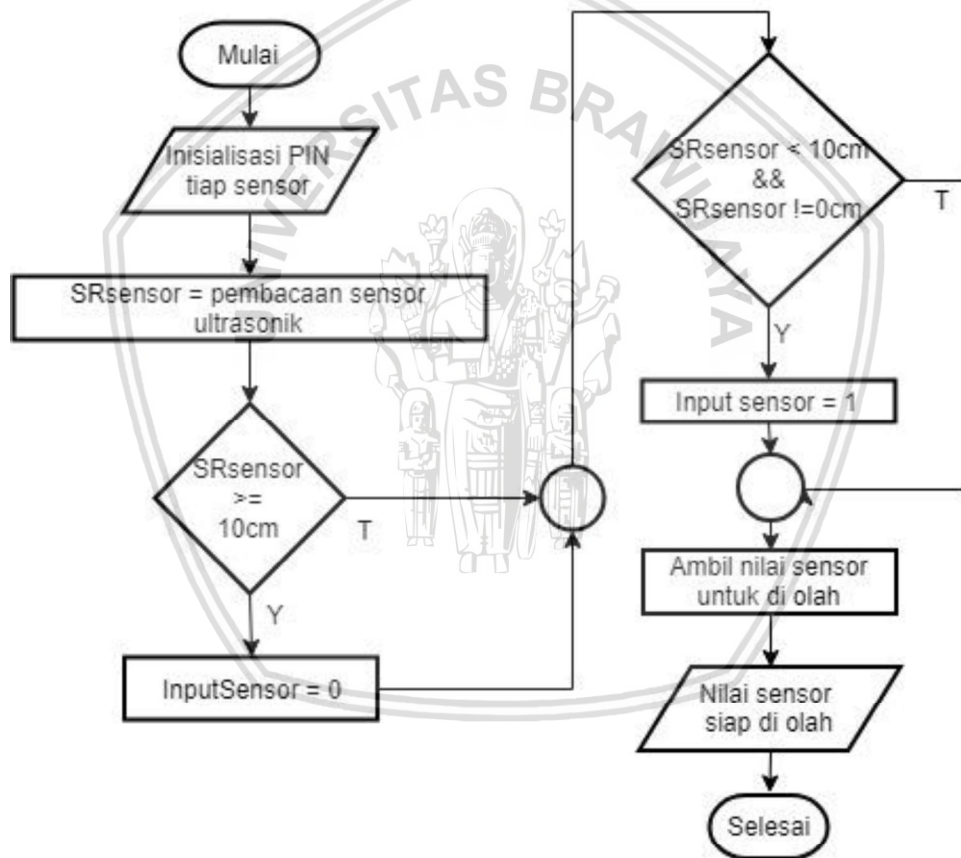
5.1.3 Perancangan Perangkat Lunak

Pada perancangan perangkat lunak pembahasannya dibagi menjadi dua bagian, yaitu perancangan pada perangkat lunak sebagai menentukan pengambilan data sensor pada mikrokontroler dan perancangan teknik encoding digital.

5.1.3.1 Perancangan Pengambilan Data sensor

Proses perancangan perangkat lunak bertujuan mengambil data sensor ditunjukkan pada Gambar 5.5, hal tersebut dimaksudkan sebagai penentu hasil pembacaan data sensor yang akan diolah untuk dilakukan penentuan gerakan dengan menggunakan teknik encoding digital, karena mikrokontroler pada

dasarnya secara terus menerus akan melakukan pembacaan nilai sensor. Sistem tersebut dimulai dengan melakukan inisialisasi pin dari masing-masing sensor pada Arduino IDE sebagai pembeda antara *input* dan *output*. Nilai pembacaan sensor tersebut akan dimasukkan kedalam variabel *SRsensor*, selanjutnya nilai variabel *SRsensor* akan di cek. Apabila *SRsensor* bernilai lebih sama dengan 10, maka variabel *input sensor* sama dengan 0, lanjut ke tahap selanjutnya apabila tidak sesuai. Apabila *SRsensor* bernilai kurang dari 10 dan *SRsensor* tidak sama dengan 0, maka variabel *input sensor* sama dengan 1, lanjut ke tahap selanjutnya apabila tidak sesuai. Selanjutnya, nilai variabel *input sensor* masing-masing sensor akan digunakan sebagai penentu pergerakan robot menggunakan teknik encoding digital, sehingga akan di dapatkan 256 kemungkinan.

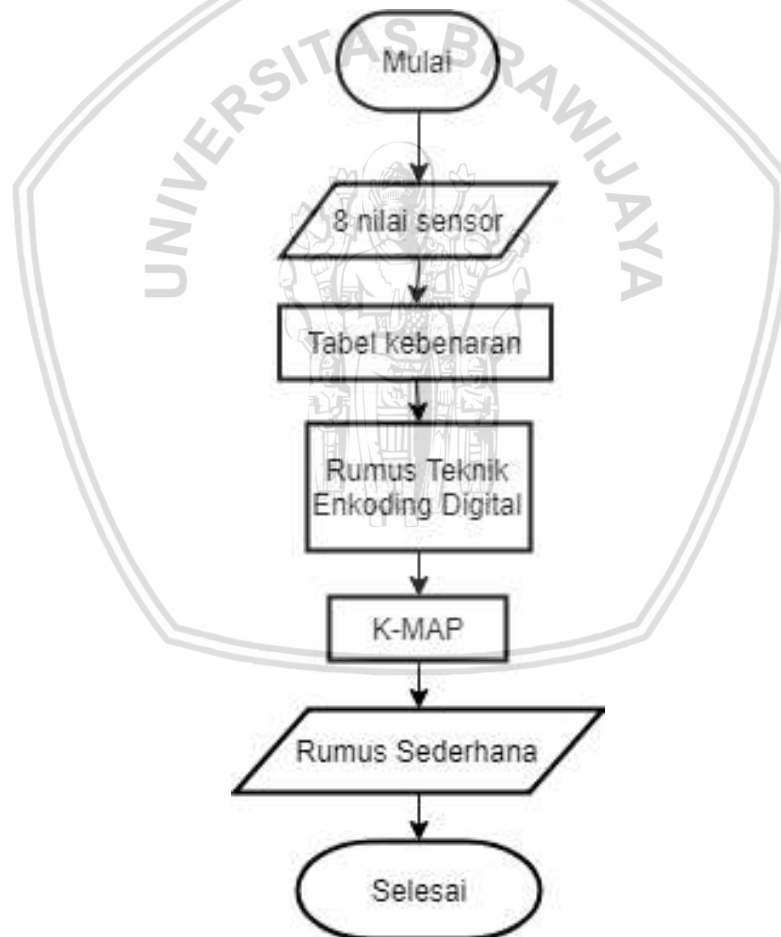


Gambar 5. 5 Diagram Alir Perancangan Perangkat Lunak Pengambilan Data Sensor

Pembacaan sensor dan proses penentuan pergerakan robot menggunakan RTOS dalam penelitian ini digunakan untuk *multitasking*. Perancangan ini dimulai dari menentukan jumlah *task* yang diperlukan, menentukan tugas tiap *task*, menentukan prioritas pada setiap *task* dan batas waktu *task* untuk dieksekusi. *Task* yang diperlukan dalam sistem ini ialah 9 karena pada sistem ini menggunakan

8 sensor HC-SR04 sebagai input data dan 1 proses, untuk prioritas pada *task* pembacaan sensor dibuat sama sebesar 1 dan untuk *task* proses memiliki prioritas yang lebih tinggi sebesar 2. Penjadwalan untuk *task* yang memiliki prioritas sama menggunakan algoritme *round-robin scheduling*. *Round robin scheduling* menggunakan *time slicing* untuk membagi mendapatkan alokasi waktu yang sama pada setiap *task*. Pembacaan sensor dalam sistem ini menggunakan *library newping.h*, dengan menggunakan *library* ini akan mempermudah dalam mengatur sensor HC-SR04 seperti inisialisasi pin dan pengaturan jarak maksimum sensor. Berdasarkan penelitian Implementasi real time pada pergerakan robot quadruped menggunakan multi sensor dan RTOS, *task* pembacaan sensor diproses terlebih dahulu, setelah itu *task* proses. (Saputro, 2018)

5.1.3.2 Perancangan Teknik Enkoding Digital

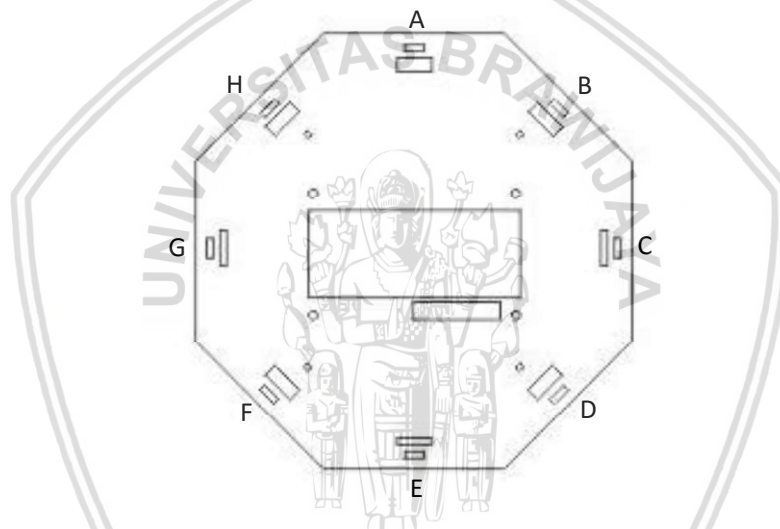


Gambar 5. 6 Diagram Alir Perancangan Teknik Enkoding Digital

Dalam teknik enkoding digital terdapat beberapa tahap-tahap sesuai dengan Gambar 5.6, sebagai masukan awal pada teknik enkoding digital adalah nilai-nilai dari hasil pembacaan sensor ultrasonik HC-SR04. Hasil pembacaan sensor

tersebutlah yang selanjutnya akan menjadi fitur-fitur untuk mempengaruhi penentuan arah pergerakan robot. Proses dimulai dari membuat tabel kebenaran dari 256 kemungkinan dari sensor terhadap gerakan robot yang dapat dilakukan. Tahap selanjutnya adalah membuat rumus untuk masing-masing gerakan dengan menggunakan teknik encoding digital. Rumus yang didapatkan tersebut dapat di sederhanakan pada tahap selanjutnya yaitu K-MAP. Sehingga didapatkan rumus teknik encoding digital setiap gerakan.

Untuk mempermudah perancangan teknik encoding digital dilakukan penamaan sensor untuk masing-masing sensor serta nama pergerakan yang dapat di lihat pada Gambar 5.7 dan Tabel 5.2, sehingga di dapatkan tabel kebenaran sensor pada LAMPIRAN A.



Gambar 5. 7 Penamaan Sensor

Tabel 5. 2 Penamaan Arah Gerak Robot

Output A	Pergerakan Robot kearah A
Output B	Pergerakan Robot kearah B
Output C	Pergerakan Robot kearah C
Output D	Pergerakan Robot kearah D
Output E	Pergerakan Robot kearah E
Output F	Pergerakan Robot kearah F
Output G	Pergerakan Robot kearah G
Output H	Pergerakan Robot kearah H

Contoh teknik encoding digital untuk melakukan penentuan rumus teknik encoding digital *Output A*. Berdasarkan tabel kebenaran pada LAMPIRAN A,

didapatkan bahwa gerakan *Output A* akan dilakukan jika input bernilai pada tabel 5.3.

Tabel 5. 3 Tabel Input yang memiliki Output A

NO	Input							
	A	B	C	D	E	F	G	H
9	0	0	0	0	1	0	0	0
13	0	0	0	0	1	1	0	0
19	0	0	0	1	0	0	1	0
21	0	0	0	1	0	1	0	0
23	0	0	0	1	0	1	1	0
25	0	0	0	1	1	0	0	0
27	0	0	0	1	1	0	1	0
29	0	0	0	1	1	1	0	0
31	0	0	0	1	1	1	1	0
35	0	0	1	0	0	0	1	0
37	0	0	1	0	0	1	0	0
39	0	0	1	0	0	1	1	0
43	0	0	1	0	1	0	1	0
45	0	0	1	0	1	1	0	0
47	0	0	1	0	1	1	1	0
51	0	0	1	1	0	0	1	0
53	0	0	1	1	0	1	0	0
55	0	0	1	1	0	1	1	0
59	0	0	1	1	1	0	1	0
61	0	0	1	1	1	1	0	0
63	0	0	1	1	1	1	1	0

Dari tabel kebenaran tersebut didapat bahwa Output A memiliki rumus teknik encoding digital

$$\begin{aligned}
 \text{Output A} = & A'B'C'D'EF'G'H' + A'B'C'D'EFG'H' + A'B'C'DE'F'GH' + A'B'C'DE'FG'H' + \\
 & A'B'C'DE'FGH' + A'B'C'DEF'G'H' + A'B'C'DEF'GH' + A'B'C'DEFG'H' + \\
 & A'B'C'D'EFGH' + A'B'CD'E'F'GH' + A'B'CD'E'FG'H' + A'B'CD'E'FGH' + \\
 & A'B'CD'EF'GH' + A'B'CD'EFG'H' + A'B'CD'EFGH' + A'B'CDE'F'GH' + \\
 & A'B'CDE'FG'H' + A'B'CDE'FGH' + A'B'CDEF'GH' + A'B'CDEFG'H' + \\
 & A'B'CDEFGH'
 \end{aligned}$$

Rumus teknik encoding digital tersebut dapat disederhanakan dengan KMAP Tabel 5.4 sebagai berikut.

Tabel 5. 4 K-MAP Pergerakan Robot A

	00 00	00 01	00 11	00 10	01 10	01 11	01 01	01 00	11 00	11 01	11 11	11 10	10 10	10 11	10 01	10 00
00 00									1							1
00 01				1	1			1	1			1	1			1
00 11				1	1			1	1			1	1			
00 10				1	1			1	1			1	1			
01 10																
01 11																
01 01																
01 00																
11 00																
11 01																
11 11																
11 10																
10 10																
10 11																
10 01																
10 00																

Dengan menggunakan KMAP rumus teknik enkoding digital *Output A* dapat disederhanakan,

$$\text{Output A} = A'B'DE'GH' + A'B'CE'GH' + A'B'EFG'H' + A'B'DFG'H' + A'B'CFG'H' + A'B'DEGH' + A'B'CEGH' + A'B'C'EF'G'H'$$

Langkah yang sama dilakukan untuk mencari rumus teknik enkoding digital *Output B, C, D, E, F, G, dan H*. Sehingga di dapat rumus teknik enkoding digital untuk setiap gerakan sebagai berikut.

$$\text{Output } B = A'B'C'DE'G + A'B'C'DE'F'H + A'B'C'DE'FH + A'B'C'E'FGH' + A'B'C'EG + A'B'C'EF'G'H + A'B'C'EFG'H + A'B'C'D'FG'H'$$

$$\text{Output } C = AB'C'D'F + AB'C'D'EF' + A'B'C'D'E'G + A'B'C'D'E'FH + A'B'C'D'EFH + A'B'C'D'EF'H$$

$$\text{Output } D = BC'D'E'G + BC'D'E'F + B'C'D'E'F'H + AB'C'D'E'F + AB'C'D'E'F'GH'$$

$$\text{Output } E = AB'C'D'E'F'G' + ACD'E'F'G + ACD'E'F'G'H + BC'D'E'F'G + BC'D'E'F'G'H + ABC'D'E'F'G' + A'CD'E'F'G + A'CD'E'F'G'H$$

$$\text{Output } F = A'CE'F'G'H + A'B'DE'F'G'H + A'BDE'F'G'H + A'BCD'E'F'G' + ACE'F'G' + AB'C'DE'F'G' + ABC'DE'F'G' + BC'D'E'F'G'H'$$

$$\text{Output } G = A'CE'F'G'H' + A'BC'DE'F'G'H' + A'BEF'G'H' + AEF'G'H' + ABDE'F'G'H' + AB'DE'F'G'H'$$

$$\text{Output } H = A'BEF'G'H' + A'B'CD'EF'G'H' + A'BFG'H' + A'CFG'H' + A'B'DF'G'H'$$

Terlihat pada rumus teknik encoding digital masing-masing gerakan memiliki rumus yang sederhana. Sehingga dari 256 kemungkinan yang terjadi pada sensor dapat dilakukan komputasi dengan rumus tersebut.

5.2 Implementasi Sistem

Pada tahap ini, implementasi sistem merupakan tahapan yang berfungsi merealisasikan dan membuat sistem berdasarkan keseluruhan perancangan yang telah dilakukan sebelumnya. Pada implementasi sistem memaparkan satu per satu secara detail mengenai implementasi robot, implementasi perangkat keras dan implementasi perangkat lunak.

5.2.1 Implementasi Robot *Quadruped*

Dalam mengimplementasikan sistem ini merujuk pada perancangan pada subbab 5.1.1 yakni menggunakan *acrylic* dengan masing-masing bagian memiliki ukurannya. Hasil perancangan desain selanjutnya dilakukan pemotongan pada bahan *acrylic*. Setelah setiap bagian *acrylic* selesai, dilanjutkan dengan merangkai setiap bagian menjadi bentuk sesuai perancangan desain beserta servo ax-12.

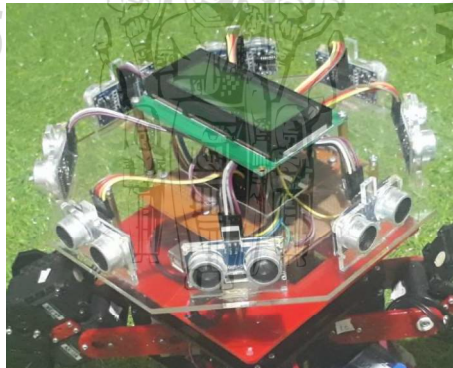
Setelah setiap bagian robot selesai di bentuk, dilakukan pemasangan komponen-komponen yang terdiri dari sensor ultrasonik HC-SR04, Arduino Mega, OpenCM9.04, dan ekspansi OpenCM 485. Hasil implementasi Robot *Quadruped* dan peletakan komponen-komponen elektronnik yang digunakan ditunjukkan pada Gambar 5.8.



Gambar 5. 8 Implementasi Robot Quadruped

5.2.2 Implementasi Perangkat Keras

Pada tahap ini menjelaskan mengenai proses pengimplementasian perangkat keras termasuk komponen elektronik antara lain Arduino Mega, Sensor Ultrasonik HC-SR04, OpenCM9.0, ekspansi OpenCM485, dan Servo Ax-12. Seluruhan komponen-komponen tersebut dirangkai satu dengan yang lain menggunakan PCB sesuai dengan perancangan yang dijelaskan pada subbab 5.1.2



Gambar 5. 9 Implementasi Delapan Sensor Ultrasonik HC-SR04

Pada Gambar 5.9 menunjukkan hasil implementasi delapan sensor ultrasonik HC-SR04 dengan arduino mega. Sensor Ultrasonik HC-SR04 telah di rangkai sesuai dengan perancangannya dan diletakkan dibagian atas robot mengarah ke sekeliling robot yang di hubungkan melalui kabel *jumper* ke Arduino Mega yang di letakkan dibagian tengah robot, di bawah sensor ultrasonik.



Gambar 5. 10 Implemntasi Servo AX-12, OpenCM9.04 dan Ekspansi OpenCM485

Berdasarkan perancangan kaki robot dengan servo AX-12, ekspansi OpenCM485, dan OpenCM9.04 yang telah terhubung Arduino Mega yang telah dijelaskan pada subbab 5.1.2, hasil dari implementasi pada Gambar 5.10.

5.2.3 Implementasi Perangkat lunak

Implementasi perangkat lunak menjelaskan mengenai proses realisasi program sebagai Implementasi Teknik Ekoding Digital Pembacaan Sensor Ultrasonik Untuk Memetaan Keputusan Aksi Robot *Quadruped* ini sesuai dengan perancangan yang telah dilakukan pada subbab 5.1.3. Dalam melakukan implementasi prangkat lunak ini seluruh proses pemrograman dilakukan pada Arduino IDE. Program terlebih dahulu dilakukan inialisasi *library* yang bertujuan untuk mempermudah pemrograman beberapa fungsi tertentu. Adapun library digunakan ditunjukkan pada Tabel 5.5.

Tabel 5. 5 Inialisasi Library

Baris	Kode Program
1	<code>#include <NewPing.h></code>
2	<code>#include <Arduino_FreeRTOS.h></code>

5.2.3.1 Implementasi Kode Program Pengambilan Data Sensor

Dalam implementasi kode program pengambilan data sensor berdasarkan perancangan perangkat keras dan perancangan perangkat lunak diperlukan inialisasi variabel dan konfigurasi pin sensor. Tahapan implementasi ini diawali dengan membuat *task-task* yang akan digunakan dengan sistem RTOS. Dalam sistem ini terdiri dari 8 sensor dan 1 proses pengolahan data, maka task yang digunakan berjumlah 9 task yaitu *TaskUS1*, *TaskUS2*, *TaskUS3*, *TaskUS4*, *TaskUS5*, *TaskUS6*, *TaskUS7*, *TaskUS8*, dan *Taskus9*. Pada tabel 5.6, baris ke-1 hingga ke-3 sebagai inialisasi pin untuk Sensor Ultrasonik HC-SR04, baris ke-7 sebagai inialisasi variabel untuk pengambilan data, baris ke-9 sebagai inialisai

method/task baru bernama *TaskUS1*, baris ke-14 sebagai inisialisasi komunikasi serial di 115200, baris ke-15 sebagai awal dari perulangan, baris ke-18 dan baris ke-19 sebagai inisialisasi *vTaskDelay* dan inisialisasi variabel *SRsensor* sama dengan nilai sensor pada pin 0 yaitu pin 34 35, baris ke-20 hingga ke-28 merupakan proses penentuan jika nilai *SRsensor* lebih besar sama dengan 10 cm maka nilai *IA* (*Input A*) sama dengan 0, jika nilai *SRsensor* kurang dari 10 cm dan *SRsensor* tidak sama dengan 0 maka nilai *IA* sama dengan 1.

Tabel 5. 6 Kode Program Implementasi Kode Program Pengambilan Data Sensor

Baris	Kode Program
1	//pin Sensor Ultrasonik HC-SR04
2	NewPing sonar[SONAR_NUM] = {
3	NewPing(34, 35, MAX_DISTANCE)
4	};
5	
6	//inisialisasi variabel untuk pengambilan data
7	bool IA=0;
8	
9	void TaskUS1(void *pvParameters)
10	{
11	(void) pvParameters;
12	// initialize serial communication at 115200 bits per
13	second
14	Serial.begin(115200);
15	for (;;)
17	{
18	vTaskDelay(100 / portTICK_PERIOD_MS);
19	int SRsensor = sonar[0].ping_cm();
20	if (SRsensor >= 10)
21	{
22	IA = 0;
23	}
24	if (SRsensor < 10 && SRsensor != 0)
25	{
26	IA = 1;
27	}}}

5.2.3.2 Implementasi Teknik Enkoding Digital

Pada Tahap Implementasi Teknik Enkoding Digital ini sebagai realisasi kode program saat melakukan pengambilan keputusan aksi robot *quadruped* untuk bergerak menjauhi halangan.

Tabel 5. 7 Kode Program Inisialisasi Variabel Teknik Enkoding Digital

Baris	Kode Program
1	bool A, B, C, D, E, F, G, H;
2	bool IA=0, IB=0, IC=0, ID=0, IE=0, IF=0, IG=0, IH=0;

Berdasarkan Tabel 5.7, tabel tersebut menunjukkan proses inialisasi tipe data dan variabel yang digunakan untuk pemrograman Teknik Enkoding Digital. Kode program tersebut diinisialisasi sebagai variabel global sehingga variabel tersebut dapat digunakan di semua fungsi yang menggunakan variabel tersebut.

Tabel 5. 8 Kode Program Implementasi Teknik Enkoding Digital

Baris	Kode Program
1	A = (!IA && !IB && ID && !IE && IG && !IH)
2	(!IA && !IB && IC && !IE && IG && !IH)
3	(!IA && !IB && IE && IF && !IG && !IH)
4	(!IA && !IB && ID && IF && !IG && !IH)
5	(!IA && !IB && IC && IF && !IG && !IH)
6	(!IA && !IB && ID && IE && IG && !IH)
7	(!IA && !IB && IC && IE && IG && !IH)
8	(!IA && !IB && !IC && IE && !IF && !IG && !IH);
9	
10	B = (!IA && !IB && !IC && ID && !IE && IG)
11	(!IA && !IB && !IC && ID && !IE && !IF && IH)
12	(!IA && !IB && !IC && ID && !IE && IF && IH)
13	(!IA && !IB && !IC && !IE && IF && IG && !IH)
14	(!IA && !IB && !IC && IE && IG)
15	(!IA && !IB && !IC && IE && !IF && !IG && IH)
16	(!IA && !IB && !IC && IE && IF && !IG && IH)
17	(!IA && !IB && !IC && !ID && IF && !IG && !IH);
18	
19	C = (IA && !IB && !IC && !ID && IF)
20	(IA && !IB && !IC && !ID && IE && !IF)
21	(!IA && !IB && !IC && !ID && !IE && IG)
22	(!IA && !IB && !IC && !ID && !IE && IF && IH)
23	(!IA && !IB && !IC && !ID && IE && IF && IH)
24	(!IA && !IB && !IC && !ID && IE && !IF && IH);
25	
26	D = (IB && !IC && !ID && !IE && IG)
27	(IB && !IC && !ID && !IE && IF)
28	(!IB && !IC && !ID && !IE && !IF && IH)
29	(IA && !IB && !IC && !ID && !IE && IF)
30	(IA && !IB && !IC && !ID && !IE && !IF && IG && !IH);
31	
32	E = (IA && !IB && !IC && !ID && !IE && !IF && !IG)
33	(IA && IC && !ID && !IE && !IF && IG)
34	(IA && IC && !ID && !IE && !IF && !IG && IH)
35	(IB && !IC && !ID && !IE && !IF && IG)
36	(IB && !IC && !ID && !IE && !IF && !IG && IH)
37	(IA && IB && !IC && !ID && !IE && !IF && !IG)
38	(!IA && IC && !ID && !IE && !IF && IG)
39	(!IA && IC && !ID && !IE && !IF && !IG && IH);
40	
41	F = (!IA && IC && !IE && !IF && !IG && IH)
42	(!IA && !IB && ID && !IE && !IF && !IG && IH)
43	(!IA && IB && ID && !IE && !IF && !IG && IH)
44	(!IA && IB && IC && !ID && !IE && !IF && !IG)
45	(IA && IC && !IE && !IF && !IG)
46	(IA && !IB && !IC && ID && !IE && !IF && !IG)
47	(IA && IB && !IC && ID && !IE && !IF && !IG)
48	(IB && !IC && !ID && !IE && !IF && !IG && !IH);

Baris	Kode Program
49	
50	G = (!IA && IC && !IE && !IF && !IG && !IH)
51	(!IA && IB && !IC && ID && !IE && !IF && !IG && !IH)
52	
53	(!IA && IB && IE && !IF && !IG && !IH)
54	(IA && IE && !IF && !IG && !IH)
55	(IA && IB && ID && !IE && !IF && !IG && !IH)
56	(IA && !IB && ID && !IE && !IF && !IG && !IH);
57	
58	H = (!IA && IB && IE && !IF && !IG && !IH)
59	(!IA && !IB && IC && !ID && IE && !IF && !IG && !IH)
60	
61	(!IA && IB && IF && !IG && !IH)
62	(!IA && IC && IF && !IG && !IH)
63	(!IA && !IB && ID && !IF && !IG && !IH);
64	

Rumus teknik enkoding digital untuk setiap gerakan yang telah di dapatkan pada subbab 5.1.3.2 dibuat dalam kode program pada Tabel 5.8. Pada rumus tersebut menggunakan gerbang logika *AND*, *OR*, dan *NOT*. Gerbang logika *AND* dalam kode program di tuliskan "&&", gerbang logika *OR* dalam kode program dituliskan "||", dan gerbang logika *NOT* dalam kode program di tuliskan "!".

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini pengujian dan analisis membahas mengenai proses pengujian serta menganalisis hasil dari beberapa pengujian yang dilakukan berdasarkan sistem yang telah dibuat. Pengujian ini bertujuan untuk mengetahui semua kebutuhan yang diharapkan telah sesuai yang di butuhkan oleh sistem. Pengujian yang dilakukan yakni menguji sensor ultrasonik HC-SR04, menguji waktu Komputasi Teknik Enkoding Digital dan menguji akurasi hasil klasifikasi Teknik Enkoding Digital.

6.1 Pengujian Sensor Ultrasonik HC-SR04

Sensor ultrasonik HC-SR04 merupakan sensor utama dalam sistem ini, yang berfungsi untuk membaca halangan di sekitar robot. Pada pengujian sensor Ultrasonik HC-SR04 ini akan dilakukan dengan melakukan pembacaan jarak sensor terhadap halangan kemudian nilai akan dibandingkan dengan alat ukur jarak yang sebenarnya.

6.1.1 Tujuan Pengujian

Tujuan dari pengujian ini adalah mengetahui seberapa tingkat keakuratan pada sensor ultrasonik HC-SR04 dalam membaca jarak suatu halangan. Dengan membandingkan nilai antara yang dibaca oleh sensor dengan nilai jarak pada alat ukur, sehingga akan didapat selisih nilai dari pembacaan sensor yang kemudian dapat diketahui nilai error dari sensor ultrasonik HC-SR04 yang diterapkan.

6.1.2 Prosedur Pengujian

Prosedur untuk menguji sensor ultrasonik HC-SR04 sebagai berikut :

1. Menghubungkan antara mikrokontroler Arduino Mega dengan laptop.
2. Meng-upload kode program dari sensor ultrasonik HC-SR04.
3. Memberikan halangan di depan sensor ultrasonik HC-SR04 dengan jarak tertentu berdasarkan alat ukur.
4. Melihat nilai jarak pada sensor ultrasonik HC-SR04 pada serial monitor.
5. Mengamati hasil nilai jarak dari sensor ultrasonik HC-SR04 pada serial monitor dan membandingkan antara nilai yang terbaca tersebut dengan nilai jarak dari alat ukur, kemudian menghitung besarnya error pembacaan sensor dengan cara mengkonversi nilai jarak dari alat ukur dan nilai jarak dari sensor ultrasonik HC-SR04.
6. Nilai dari alat ukur dan sensor ultrasonik HC-SR04 kemudian selisihnya diambil sehingga dapat dilakukan perhitungan presentase error dalam bentuk nilai desimal

Untuk mengukur persentase error dapat dilakukan dengan menggunakan persamaan 6.1 berikut :

$$\text{Persentase error} = \frac{\text{Selisih nilai pembacaan}}{\text{Pembacaan alat ukur}} \times 100\% \quad (6.1)$$

Untuk menghitung selisih nilai pembacaan sensor dan alat ukur dengan menggunakan Persamaan 6.2

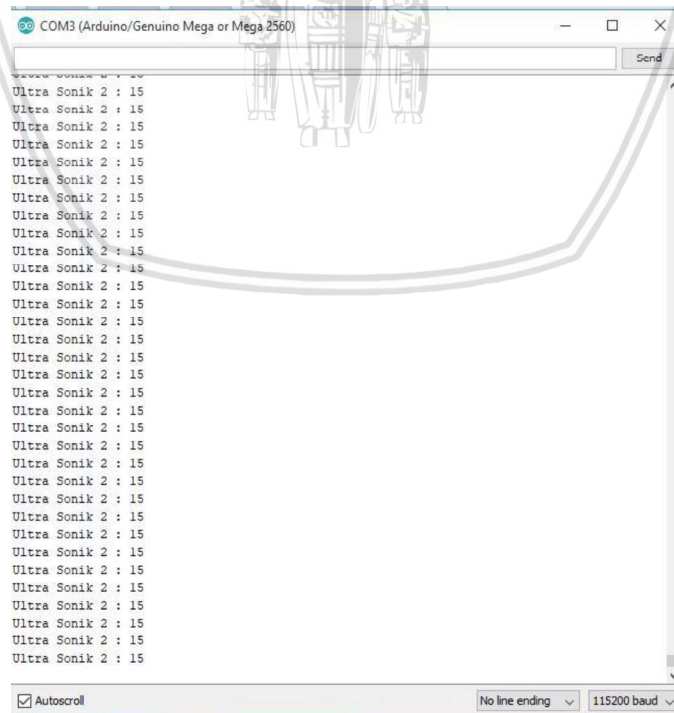
$$\text{Selisih nilai pembacaan} = \text{alat ukur} - \text{pembacaan sensor} \quad (6.2)$$

6.1.3 Hasil dan Analisis Pengujian



Gambar 6. 1 Pengujian Menggunakan Alat Ukur

Sesuai dengan prosedur analisis, pengujian dengan alat ukur pada Gambar 6.1 akan di bandingkan dengan pengujian dengan sensor yang ditampilkan pada serial monitor pada Gambar 6.2.



Gambar 6. 2 Tampilan Serial Monitor Pengujian Menggunakan Sensor

Tabel 6. 1 Hasil Pengujian Sensor Ultrasonik HC-SR04 1

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 2 Hasil Pengujian sensor Ultrasonik HC-SR04 2

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 3 Hasil Pengujian Sensor Ultrasonik HC-SR04 3

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 4 Hasil Pengujian Sensor Ultrasonik HC-SR04 4

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 5 Hasil Pengujian Sensor Ultrasonik HC-SR04 5

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 6 Hasil Pengujian Sensor Ultrasonik HC-SR04 6

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	16	1	0,07%
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 7 Hasil Pengujian Sensor Ultrasonik HC-SR04 7

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%
2	13	13	0	0,00%
3	10	10	0	0,00%

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 8 Hasil Pengujian Sensor Ultrasonik HC-SR04 8

No	Jarak Pada Alat ukur	Jarak Pada Sensor Ultrasonik HC-SR04	Selisih	Error
1	15	15	0	0,00%
2	13	13	0	0,00%
3	10	10	0	0,00%
4	8	8	0	0,00%
5	5	5	0	0,00%
6	3	3	0	0,00%
Rata-rata				0,00%

Tabel 6. 9 Hasil Pengujian Semua Sensor Ultrasonik

Sensor Ultrasonik Ke-	Rata-Rata persentase Error
1	0,00%
2	0,00%
3	0,00%
4	0,00%
5	0,00%
6	0,00%
7	0,00%
8	0,00%
Rata-rata	0,00%

Berdasarkan Persamaan (6.1) untuk menentukan persentase error pada setiap pengujian sensor ultrasonik HC-SR04 yang di tunjukan pada Tabel 6.1, Tabel 6.2, Tabel 6.3, Tabel 6.4, Tabel 6.5, Tabel 6.6, Tabel 6.7, Tabel 6.8, diperoleh rata-rata error pada Tabel 6.9 sebesar 0,0%. Contoh perhitungan persentase error pada pengujian Sensor Ultrasonik HC-SR04 1 :

Jarak pada alat ukur = 15cm

Jarak pada sensor ultrasonik HC-SR04 = 15cm

Selisih nilai pembacaan = pembacaan alat ukur – pembacaan sensor

$$\text{Selisih nilai pembacaan} = 15\text{cm} - 15\text{cm}$$

$$\text{Selisih nilai pembacaan} = 0$$

$$\text{Persentase error} = \frac{\text{Selisih nilai pembacaan}}{\text{pembacaan alat ukur}} \times 100\%$$

$$\text{Persentase error} = \frac{0}{15} \times 100\%$$

$$\text{Persentase error} = 0\%$$

Untuk nilai rata-rata error keseluruhan pengujian dapat dihitung sebagai berikut:

$$\text{Rata-rata error} = \frac{\text{Jumlah persentase error}}{\text{Jumlah pengujian}}$$

$$\begin{aligned} \text{Rata-rata error} \\ &= \frac{0,00\% + 0,00\% + 0,00\% + 0,00\% + 0,00\% + 0,00\% + 0,00\% + 0,00\%}{8} \end{aligned}$$

$$\text{Rata-rata error} = 0,00\%$$

Rata-rata error sensor ultrasonik HC-SR04 sebesar 0%, sehingga akurasi dari pembacaan sensor ultrasonik HC-SR04 sangatlah baik. Walaupun error pembacaan sensor sebesar 0%, jarak yang mampu dibaca sensor ultrasonik HC-SR04 yaitu 3cm hingga 200cm.

6.2 Pengujian Waktu Komputasi Teknik Enkoding Digital

6.2.1 Tujuan Pengujian

Tujuan dari dilakukannya pengujian waktu komputasi teknik enkoding digital adalah untuk mengetahui lama waktu yang dibutuhkan sistem untuk melakukan proses pemetaan arah gerak menggunakan teknik enkoding digital, hal ini diperlukan sehingga diketahui performansi dari sistem yang telah dibuat.

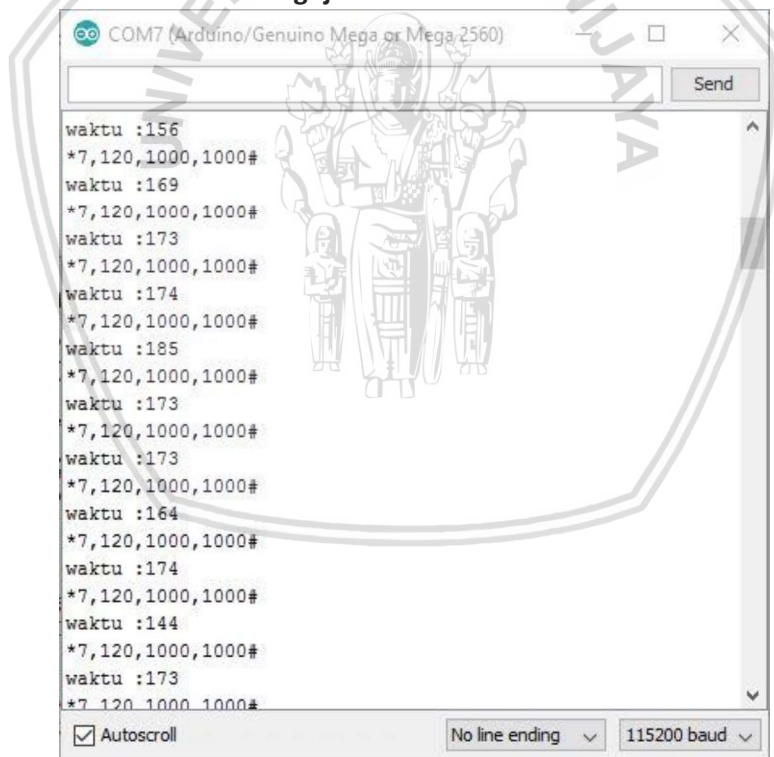
6.2.2 Prosedur Pengujian

Untuk pengujian waktu komputasi pemrosesan sistem, dilakukan dengan cara mengukur waktu komputasi ketika program teknik enkoding digital dimulai dan waktu komputasi ketika program teknik enkoding digital selesai dan dalam satu siklus sebanyak 25 kali pengujian. Pengujian dilakukan sebanyak 2 kali dengan pemberian halangan pada salah satu sensor berjarak 7cm dan 20cm. Setelah di ketahui waktu komputasi di mulai dan selesai, maka dapat di ketahui waktu komputasi program tersebut dari selisih waktu selesai dan waktu mulai. Pengujian tersebut dilakukan dengan menerapkan fungsi dari millis() (milisecond) pada kode program arduino mega pada sistem ini. Fungsi milis tersebut dihitung dalam satuan milisecond, dengan fungsi millis() yang di terapkan pada arduino ditunjukkan pada tabel 6.10

Tabel 6. 10 Kode Program Menghitung Waktu Komputasi

Baris	Kode Program
1	unsigned long sebelum, sesudah, waktu;
2	
3	void loop()
4	{
5	
6
7	//pengambilan keputusan gerak dengan teknik enkoding digital
8
9	
10	sesudah = millis();
11	waktu = sesudah - sebelum;
12	Serial.print("waktu = ");
13	Serial.println(waktu);
14	sebelum = millis();
15	}

6.2.3 Hasil dan Annalisis Pengujian

**Gambar 6. 3 Tampilan Waktu Komputasi pada Serial Monitor**

Sesuai dengan prosedur analisis Gambar 6.3 menunjukkan hasil waktu komputasi implementasi teknik enkoding digital pada robot Quadruped.

**Tabel 6. 11 Hasil Pengujian Waktu Komputasi Dengan Salah satu sensor
berjarak 7cm**

No	Nama Pengujian	Waktu Komputasi (ms)
1	Pengujian ke-1	179
2	Pengujian ke-2	188
3	Pengujian ke-3	110
4	Pengujian ke-4	194
5	Pengujian ke-5	179
6	Pengujian ke-6	188
7	Pengujian ke-7	110
8	Pengujian ke-8	146
9	Pengujian ke-9	185
10	Pengujian ke-10	155
11	Pengujian ke-11	176
12	Pengujian ke-12	187
13	Pengujian ke-13	169
14	Pengujian ke-14	172
15	Pengujian ke-15	156
16	Pengujian ke-16	169
17	Pengujian ke-17	173
18	Pengujian ke-18	174
19	Pengujian ke-19	185
20	Pengujian ke-20	173
21	Pengujian ke-21	173
22	Pengujian ke-22	164
23	Pengujian ke-23	174
24	Pengujian ke-24	144
25	Pengujian ke-25	173
Rata-rata		167,84

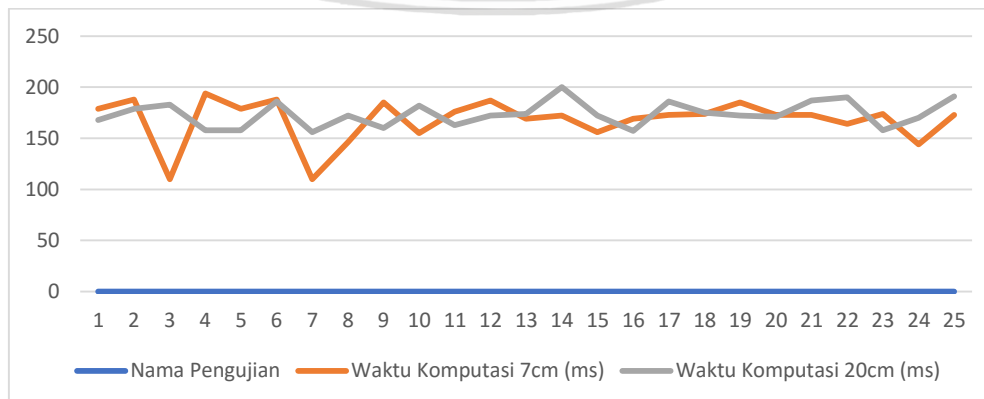
Berdasarkan hasil pengujian waktu komputasi dengan salah satu sensor di beri jarak 7cm pada Tabel 6.11 yang dilakukan sebanyak 25 kali, waktu komputasi sistem untuk mendapatkan keputusan gerak robot menggunakan teknik encoding digital dengan rata-rata waktu sebesar 167,84 ms atau sekitar 0,16 detik.

**Tabel 6. 12 Hasil Pengujian Waktu Komputasi dengan Salah Satu Sensor
Berjarak 20cm**

No	Nama Pengujian	Waktu Komputasi (ms)
1	Pengujian ke-1	168
2	Pengujian ke-2	179
3	Pengujian ke-3	183
4	Pengujian ke-4	158
5	Pengujian ke-5	158

No	Nama Pengujian	Waktu Komputasi (ms)
6	Pengujian ke-6	186
7	Pengujian ke-7	156
8	Pengujian ke-8	172
9	Pengujian ke-9	160
10	Pengujian ke-10	182
11	Pengujian ke-11	163
12	Pengujian ke-12	172
13	Pengujian ke-13	174
14	Pengujian ke-14	200
15	Pengujian ke-15	172
16	Pengujian ke-16	157
17	Pengujian ke-17	186
18	Pengujian ke-18	175
19	Pengujian ke-19	172
20	Pengujian ke-20	171
21	Pengujian ke-21	187
22	Pengujian ke-22	190
23	Pengujian ke-23	158
24	Pengujian ke-24	170
25	Pengujian ke-25	191
Rata-rata		173,6

Berdasarkan hasil pengujian waktu komputasi dengan salah satu sensor di beri jarak 20cm pada Tabel 6.12 yang dilakukan sebanyak 25 kali, waktu komputasi sistem untuk mendapatkan keputusan gerak robot menggunakan teknik enkoding digital dengan rata-rata waktu sebesar 173,6 ms atau sekitar 0,17 detik. Perbedaan antara waktu komputasi yang berjarak 7cm dan 20cm pada Gambar 6.4, yang berarti jarak mempengaruhi waktu komputasi dimana waktu komputasi yang berjarak 20cm lebih besar di bandingkan waktu komputasi yang berjarak 7cm.



Gambar 6. 4 Grafik Pengujian Waktu Komputasi

6.3 Pengujian Akurasi Hasil Klasifikasi Teknik Enkoding Digital

Implementasi Teknik Enkoder Digital Pembacaan Sensor Ultrasonik untuk Memetakan Keputusan Aksi Robot Quadruped mempunyai tujuan utama untuk memetakan arah gerak robot quadruped berdasarkan halangan disekitar robot, oleh karena itu perlu diketahui tingkat keakurasian teknik enkoding digital.

6.3.1 Tujuan Pengujian

Tujuan dilakukan pengujian ini adalah sebagai penentu nilai akurasi penggunaan teknik enkoding digital pada robot quadruped yang telah di buat.

6.3.2 Prosedur Pengujian

Prosedur pengujian akurasi Teknik Enkoding Digital dilakukan dengan meletakkan halangan di sekitar robot dan memperhatikan arah gerak robot tersebut. Untuk menentukan nilai akurasi dengan cara membandingkan arah gerak seharusnya dengan arah gerak saat pengujian. Rumus untuk menghitung akurasi ditunjukkan pada Persamaan (6.3)

$$\text{Akurasi} = \frac{\text{Total data} - \text{Data tidak sesuai}}{\text{Total data}} \times 100\% \quad (6.3)$$

6.3.3 Hasil dan Analisis Pengujian

Tabel 6. 13 Hasil Pengujian Teknik Enkoding Digital

NO	INPUT								OUTPUT		KESESUAIAN
	A	B	C	D	E	F	G	H	YANG DIINGINKAN	HASIL SISTEM	
1	0	0	0	0	0	0	0	0	Diam	DIAM	Sesuai
2	0	0	0	0	0	0	1	0	C	C	Sesuai
3	0	0	0	0	0	1	1	0	B/C	C	Sesuai
4	0	0	0	0	1	0	0	0	A	A	Sesuai
5	0	0	0	0	1	1	0	0	A/B	A	Sesuai
6	0	0	0	1	1	0	0	0	A/H	A	Sesuai
7	0	0	1	0	0	0	1	0	A/E	A	Sesuai
8	0	0	0	0	1	0	1	0	B	B	Sesuai
9	0	0	0	0	1	1	1	1	B/C	B	Sesuai
10	0	0	0	1	0	0	1	1	B	B	Sesuai
11	0	0	0	1	1	1	1	1	B	B	Sesuai
12	0	0	0	0	0	1	0	1	C	C	Sesuai
13	0	0	0	0	1	0	0	1	B/C	C	Sesuai
14	0	1	0	0	0	1	0	1	D	D	Sesuai
15	0	1	0	0	0	0	1	1	D/E	D	Sesuai
16	0	1	0	0	0	0	0	1	E	E	Sesuai
17	0	1	0	0	0	0	1	0	D/E	E	Sesuai
18	0	1	0	0	0	1	1	1	D	D	Sesuai
19	0	1	0	0	1	0	0	0	G/H	G	Sesuai

NO	INPUT								OUTPUT		KESESUAIAN
	A	B	C	D	E	F	G	H	YANG DIINGINKAN	HASIL SISTEM	
20	0	1	0	1	0	0	0	0	G	G	Sesuai
21	0	1	0	1	1	0	0	0	G/H	G	Sesuai
22	0	1	0	1	1	1	0	0	H	H	Sesuai
23	0	1	1	0	1	0	0	0	G/H	H	Sesuai
24	0	1	1	0	1	1	0	0	H	H	Sesuai
25	0	0	1	1	1	1	0	0	A/H	H	Sesuai
26	0	1	0	1	0	0	0	1	F	F	Sesuai
27	0	1	1	0	0	0	0	0	F/G	F	Sesuai
28	0	1	1	0	0	0	0	1	E/F	F	Sesuai
29	0	1	1	0	0	0	1	0	E	E	Sesuai
30	0	1	1	0	0	0	1	1	E	E	Sesuai
31	0	1	1	0	0	1	0	1	Diam	Diam	Sesuai
32	0	1	1	1	0	0	0	1	F	F	Sesuai
33	1	0	0	0	0	0	0	0	E	E	Sesuai
34	1	0	0	0	0	0	0	1	D/E	E	Sesuai
35	1	0	0	1	1	0	0	0	G	G	Sesuai
36	1	1	0	0	0	1	0	0	D	D	Sesuai
37	1	1	0	0	1	0	0	1	Diam	Diam	Sesuai
38	1	1	0	0	1	0	1	0	Diam	D	Tidak Sesuai
39	1	1	0	0	1	0	1	1	Diam	Diam	Sesuai
40	1	1	0	0	1	1	0	0	Diam	Diam	Sesuai
41	0	0	0	0	0	1	0	0	B	B	Sesuai
42	0	0	0	0	1	0	0	0	A	A	Sesuai
43	0	0	0	1	0	0	0	0	H	H	Sesuai
44	0	0	1	0	0	0	1	1	E	E	Sesuai
45	0	1	0	0	0	0	0	0	F	F	Sesuai
46	1	0	0	0	1	0	0	1	C	C	Sesuai
47	1	0	0	0	1	0	1	0	C	C	Sesuai
48	1	1	0	0	0	1	0	1	D	D	Sesuai
49	1	1	1	0	1	0	0	0	G	G	Sesuai
50	1	1	1	1	1	1	1	1	Diam	Diam	Sesuai

Berdasarkan hasil pengujian yang ditunjukkan pada Tabel 6.13, dari jumlah 50 kali pengujian terdapat 1 arah gerak yang tidak sesuai dengan tabel kebenaran. Sehingga akurasi yang diperoleh implementasi teknik enkoding digital pembacaan sensor ultrasonik untuk memetakan keputusan aksi robot quadruped adalah 98 %. Ketidak sesuaian tersebut di sebabkan ketidak sesuaian nilai sensor saat membaca, yang di sebabkan penggunaan RTOS

BAB 7 PENUTUP

Pada bab penutup memuat penarikan kesimpulan yang berdasarkan tahap-tahap yang telah dilakukan sebelumnya. Selanjutnya pada bab penutup ini peneliti menyampaikan saran yang diharapkan sehingga dapat digunakan untuk pengembangan penelitian selanjutnya yang serupa dengan penelitian ini.

7.1 Kesimpulan

Berdasarkan rumusan masalah yang telah diajukan diawal penelitian ini dan berdasarkan hasil dari analisis pengujian yang dilakukan maka didapatkan beberapa kesimpulan sebagai berikut :

1. Tingkat akurat sensor ultrasonik HC-SR04 terhadap lingkungan sekitar yang ditempatkan di bagian atas robot quadruped dapat membaca halangan berukuran panjang 8cm dan tinggi 24cm dengan rata-rata error yakni sebesar 0,00% sehingga sensor dikatakan berkerja dengan baik karena dapat sensor membaca halangan dengan jarak 3cm hingga 400cm.
2. Penerepan teknik enkoding digital pada robot quadruped telah terapkan dengan menggunakan sensor ultrasonik untuk memetakan arah gerak robot. Dimana seluruh komponen alat yang digunakan maupun teknik enkoding digital yang diinginkan dapat diterapkan dan berjalan sesuai yang direncanakan, terbukti dengan robot quadruped dapat bergerak menjauhi halangan disekitar robot.
3. Tingkat akurasi pada sistem pengambilan keputusan robot menggunakan teknik enkoding digital dengan sensor ultrasonik untuk memetakan keputusan aksi robot quadruped yang diuji memperoleh tingkat akurasi sebesar 100%. Performansi Implementasi teknik enkoding digital pembacaan sensor ultrasonik untuk memetakan keputusan aksi robot quadruped mempunyai nilai kecepatan waktu komputasi dengan salah satu sensor di beri jarak 7cm didapatkan rata-rata sebesar 0,16 detik dari 25 kali pengujian dan salah satu sensor di beri jarak 20cm didapatkan rata-rata sebesar 0,17 detik dari 25 kali pengujian. Jarak halangan dari sensor ultrasonik mempengaruhi waktu komputasi. Semakin jauh jarak sensor dengan halangan, semakin lama waktu komputasi.

7.2 Saran

Adapun saran-saran yang dapat dijadikan acuan sebagai pengembangan penelitian ini ataupun penelitian yang serupa selanjutnya adalah sebagai berikut :

1. Menggunakan metode atau algoritma klasifikasi lain sebagai pembanding metode atau algoritma yang mempunyai tingkat akuratan yang tinggi dan waktu komputasi yang lebih cepat.
2. Menggunakan sensor yang lebih sedikit sehingga mampu menghemat penggunaan sumber daya.
3. Desain robot dan penempatan sensor yang lebih baik sehingga robot mampu mendeteksi halangan lebih baik.
4. Menerapkan Teknik Enkoding digital pada sistem lain yang sesuai dengan kebutuhan.
5. Menerapkan RTOS yang lebih baik pada sensor dan gerakan robot.



DAFTAR PUSTAKA

- Adriansyah, A. (2014). Perancangan Robot Localization. *ResearchGate*.
- Arduino. (2017). *store.arduino.com*. Diambil kembali dari Arduino UNO: <https://store.arduino.cc/usa/arduino-uno-rev3>
- Arduinolearning. (2017). *Sensor Ultrasonik HC-SR04*. Diambil kembali dari Arduinolearning: <http://arduinolearning.com/code/hc-sr04-ultrasonic-sensor-example.php>
- Awal, H. (2015). Algoritma Fuzzy Logic dan Wallfollower pada Sistem Navigasi Robot Hexapod Berbasis Mikrokontroler. *Jurnal KomTekInfo Fakultas Ilmu Komputer*.
- Electronics-Tutorials. (2017). *Priority Encoder*. Diambil kembali dari Electronics-Tutorials: https://www.electronics-tutorials.ws/combinational/comb_4.html
- Kho, D. (2015). *Pengertian Gerbang Logika Dasar dan Jenis-jenisnya*. Diambil kembali dari Teknik Elektro: <http://teknikelektronika.com/pengertian-gerbang-logika-dasar-simbol/>
- KRI, K. R. (2017). *Tentang KRI*. Diambil kembali dari KRI: <https://kontesrobotindonesia.id/tentang-kri.html>
- ROBOTIS. (2010). *OpenCM9.04*. Diambil kembali dari Robotis: <http://support.robotis.com/en/product/controller/opencm9.04.htm>
- Robotis. (2016). *User Manual Dynamixel AX-12*. Robotis.
- ROBOTIS. (2017). *OPENCM 485 EXPANSION BOARD*. Diambil kembali dari ROBOTIS: <http://www.robotis.us/opencm-485-expansion-board/>
- Rouse, M. (2016). *Robot*. Diambil kembali dari whatis.com.
- Syahwiw, M. (2013). *Panduan Mudah Simulasi dan Praktek Mikrikontroler*. Indonesia: Penerbit Andi.
- Team, E. (2015). *Parameter HC-SR04*.
- Timur, M. B., Fanany, A., & Wajiansyah, A. (2017). Desain dan Implementasi Kendali Cerdas Untuk Robot Quadpod (Berkaki Empat) - Studi Kasus Robot Pemadam Api. *Jurnal Teknologi Terpadu*.